

この資料は日本エム・テクノロジー学会員専用です。
この資料を学会員以外がコピーしたり、学会員以外に配布することを禁じます。

Copy right : M Technology Association - Japan

日本エム・テクノロジー学会事務局
〒259-1193 神奈川県伊勢原市望星台
東海大学医学部・基礎医学系
大櫛陽一

Tel: 0463-93-1121 ext. 2140

Fax: 0463-96-4301

Email: youichi@keyaki.cc.u-tokai.ac.jp



第20回日本エム・テクノロジー学会大会

予稿集

1993年9月17日～9月19日
島根県出雲市

汎用シェルと医療向けシェルの知識表現と推論機構	34
今泉幸雄、大櫛陽一	
自然語情報をデータベースに取込むための構造言語学の応用	42
マルシオ・ピチキ・ド・アマラル、里村洋一	
M言語とMedical Decision Making	46
鎌江伊三夫、笹川紀夫、林 振健、山本和子	
個人病院に於ける病院情報システムの活用	
レセプト専用機からMUMPSマシンに変更して	48
土屋喬義、土屋恭子、木村一元	
千葉大学病院医療情報システム(system-CHIBA)におけるシステム運用	52
本多正幸、里村洋一、鈴木隆弘、山崎俊司、高林克日己、 赤井暁、石井晃、中村裕義、野口昇、北田光一、清水宗	
中央検査部トータルシステムの再構築	54
中尾満、森田寛二、藤井厚男、東島正満、 翼典之、花坂志郎、平山清和	
未熟児部および産科データベースの構築とその結合	62
田中吾朗、渡辺 博、木村一元	
日本病理剖検輯報の日本語データベース	64
馬場謙介	
健診システムにおける病名と家族歴の登録サブシステムの開発	66
大櫛陽一、笹川紀夫、高橋正宏、原 寿夫	
選択必修科目履修申告システムの開発	70
斉藤成広、松木三徳、菊地良平、大櫛陽一	
SQLによるDSMデータベースへのアクセス	72
辰己岳欣	
U-MUMPSクロスシステムジャーナルによるミラーシステムの構築	74
吉村貴由、木村 隆、伊藤 徹、吉中位知郎、藤江 昭	
MとTPモニターとの統合の試み	76
今井敏雄、佐藤真美	
最新DSMの特徴と今後のプラン(実演)	80
佐藤比呂志	
逆\$ORDER関数の効用ー本院の入院診療録管理システムへの応用を例に	84
馬場謙介、石名田洋一、鈴木孝始	

第20回日本エム・テクノロジー学会大会

メインテーマ：飛翔——未来のエム・テクノロジー——

日 程：1993年9月17日～9月19日

会 場：出雲市民会館（3F）
〒693 島根県出雲市塩冶有原町2-15
TEL 0853-24-1212

大会長：山本和子（島根医科大学医学部医療情報学講座）

大会事務局：〒693 島根県出雲市塩冶町89-1
島根医科大学医療情報学講座
第20回日本エム・テクノロジー学会大会
事務局 笹川紀夫
TEL 0853-23-2111 Ext.2944
FAX 0853-25-2764
NIFTY KGH00070

【プログラム委員会】

委員長 木村一元（独協医科大学）
委員 嶋 芳成（日本ダイナシステム株式会社）
委員 上戸 隆（住友電気工業株式会社）
委員 山下芳範（福井医科大学）
委員 笹川紀夫（島根医科大学）

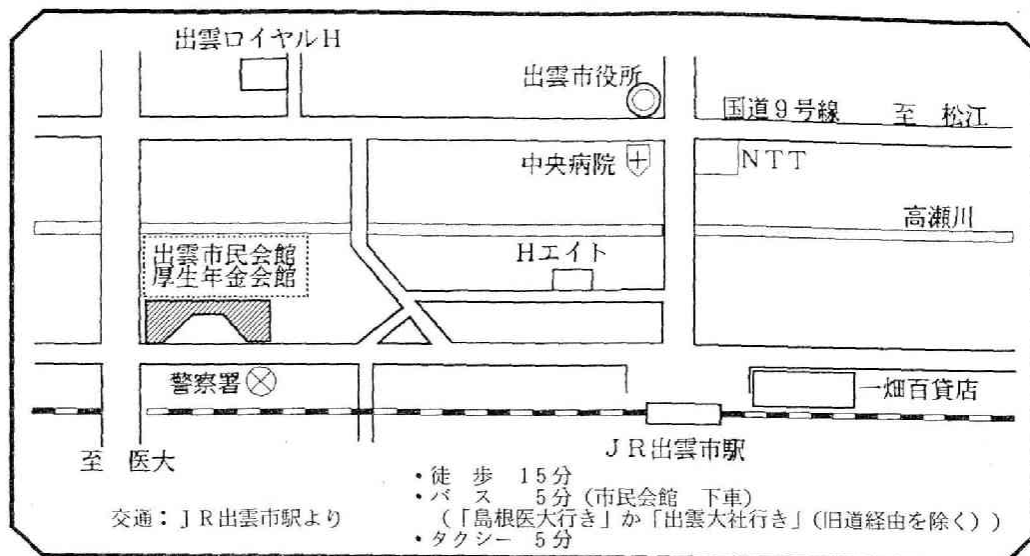
◎会場での受付 9月17日は13:00より受付を致します。
9月18, 19日は9:00より受付を致します。
尚、MTA会員で年会費未納な方は受付にてお支払下さい。

◎実演セッション 9月18日 10:00から9月19日 15:30まで行います。

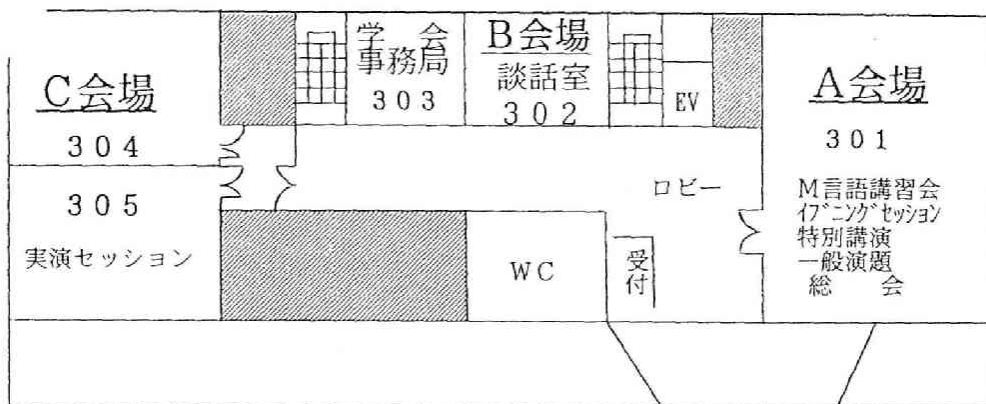
◎発表者の方へ スライド1面とOHP1台を用意致します。
一般演題の発表時間は12分（講演9分、質問3分）です。
時間厳守でお願い致します。

◎昼食 1Fに喫茶店と、隣の厚生年金会館1Fにレストランがございます。

学会場案内図



学会場見取図 (出雲市民会館 3F)



大会会場 : 出雲市民会館
(住所) 〒693 島根県出雲市塩冶有原町2-15
(電話) 0853-24-1212

【協賛各社】 (五十音順 敬称略)

- ・株式会社高崎共同計算センター
- ・住友電工システムズ株式会社
- ・ダイワボウ情報システム株式会社
- ・日本ダイナシステム株式会社
- ・日本デジタルイクイップメント株式会社
- ・日本通運株式会社

A 会場 301 (3F)

時間	17日(金)	18日(土)	19日(日)
9		学会登録受付 9:25~9:30 開会の辞 山本和子 9:45~10:00 実演セッション紹介 司会 今井敏雄(日本DEC)	9:00~9:48 一般セッションⅢ 「医学におけるMUMPS1」 座長 田久浩志(東邦大学) 沢田 薫(名古屋第二日赤)
10		10:00~11:00 一般セッションⅠ 「テクニカルセッション」 座長 嶋 芳成(日本ダイナシステム) 上戸 隆(住友電気工業)	9:48~10:36 一般セッションⅣ 「医学におけるMUMPS2」 座長 小森 優(京都大学) 今泉幸雄(SASジャパン)
11		11:00~12:00 一般セッションⅡ 「画面制御」 座長 岡田好一(筑波大学) 松本重雄(システム技研)	10:36~11:12 一般セッションⅤ 「医学におけるMUMPS3」 座長 中尾 満(大阪市立大学) 水田守秀(京都大学)
12		12:00~13:00 昼食	11:15~12:00 特別講演 Jone. R. Dumas 司会 鎌江伊三夫(島根医大)
13	学会登録受付 13:30~16:00	13:00~13:45 総会 司会 日本MTA会長 河村徹郎 (鈴鹿医療科学技術大学)	13:00~14:00 一般セッションⅥ 「MUMPSの新技术」 座長 馬場謙介(国立埼玉病院) 藤江 昭(住友電エシステムズ)
14	M言語 初級講習会 講師: 菊栗 純子 伊藤 徹 (住友電エシステムズ)	13:45~14:00 海外報告 山下芳範 14:00~15:00 特別講演 「国際化されたM言語の姿」 Prof. Richard F. Walters 司会 若井一郎 (マンブスシステム研究所)	14:00~15:00 シンポジウムⅢ 「M言語教育」 座長 河村徹郎 (鈴鹿医療科学技術大学) 世川紀夫(島根医大)
15		15:00~16:00 シンポジウムⅠ 「7*24*7M言語のJIS化について」 座長 大柳陽一(東海大学) 小松義彦(日本DEC)	15:00~15:15 閉会の辞 山本和子
16	16:00~18:30 M言語 上級講習会 講師: 嶋 芳成 鈴木 利明 (日本ダイナシステム)	16:00~17:00 シンポジウムⅡ 「M言語装備の現状と展望」 座長 山下芳範(福井医大) 木村一元(独協医大)	
17		17:00~17:30 ワークショップ 「ソフトの標準化」 座長 本多正幸(千葉大学) 林 恭平(京都府立医大)	
18	18:30~20:00 イブニングセッション 「これならわかる オブジェクト指向」 オーガナイザー 今泉 幸雄 (SASジャパン)	18:00~20:00 懇親会 厚生年金会館	玉造温泉

C 会場 304, 305 (3F)

実演セッション

[9月18日(土)10:00~9月19日(日)15:30]

参加者: ・千葉大学病院医療情報部 ・システム技研株式会社
 ・筑波大学臨床医学系 ・日本デジタルイクイップメント株式会社
 ・住友電エシステムズ株式会社 ・日本ダイナシステム株式会社

プログラム

特別講演	A会場 (301)
・「国際化されたM言語の姿」 Features for the International M Language 演者：Prof. Richard F. Walters 司会：若井一郎 (マンブスシステム研究所)	【9月18日 (土) 14:00~15:00】
・「ダウンサイジングとLANの利用」 演者：Mr. John R. Dumas 司会：鎌江伊三夫 (島根医科大学)	【9月19日 (日) 11:15~12:00】
シンポジウム	A会場 (301)
1. 「プログラム言語のJIS化について」 座長：大櫛陽一 (東海大学) ：小松義彦 (日本デジタルイクイップメント)	【9月18日 (土) 15:00~16:00】
2. 「M言語装備の現状と展望」 座長：山下芳範 (福井医科大学) ：木村一元 (独協医科大学)	【9月18日 (土) 16:00~17:00】
3. 「M言語教育」 座長：河村徹郎 (鈴鹿医療科学技術大学) ：笹川紀夫 (島根医科大学)	【9月19日 (日) 14:00~15:00】
ワークショップ	A会場 (301)
・「ソフトの標準化 ・・・システムの共有化、プログラムの共同開発は可能か」 座長：本多正幸 (千葉大学) ：林 恭平 (京都府立医科大学)	【9月18日 (土) 17:00~17:30】
日本MTA総会	A会場 (301)
司会：日本MTA会長 河村徹郎 (鈴鹿医療科学技術大学)	【9月18日 (土) 13:00~13:45】
海外活動報告	A会場 (301)
「北米MTA大会報告」 「MDC (M開発委員会) 報告」 座長：河村徹郎 (鈴鹿医療科学技術大学) 報告者：山下芳範 (福井医科大学)	【9月18日 (土) 13:45~14:00】

イブニングセッション

A会場 (301)

「これならわかるオブジェクト指向」 【9月17日(金) 18:30~20:00】
ー手続き言語時代の技術者を迷わしたオブジェクトとは何か?ー

オーガナイザー：今泉幸雄 (SASインスティテュートジャパン)

M言語講習会

A会場 (301)

1. M言語初級講習会 【9月17日(金) 13:30~16:00】

講師：菊楽純子、伊藤 徹 (住友電工システムズ)

2. M言語上級講習会 【9月17日(金) 16:00~18:30】

講師：嶋 芳成、鈴木利明 (日本ダイナシステム)

実演セッション紹介

A会場 (301)

司 会：今井 敏雄 【9月18日(土) 9:30~10:00】

(日本デジタルイクイップメント)

実演セッション

C会場(304,305)

【9月18日(土)10:00~9月19日(日)15:30】

- ・千葉大学医学部附属病院医療情報部
- ・筑波大学臨床医学系
- ・日本デジタルイクイップメント株式会社
- ・住友電工システムズ株式会社
- ・日本ダイナシステム株式会社
- ・システム技研株式会社

一般演題 第1日目 【9月18日(土)】 A会場(301)

◎一般セッションI: テクニカルセッション 【10:00~11:00】

座長: 嶋 芳成 (日本ダイナシシステム)
: 上戸 隆 (住友電気工業)

- 1-1 「パソコンM言語と他言語インターフェイス
(C言語、アセンブラ言語による外部モジュールの開発)」
○沢田 潔、木下 元一、浅井 廣、岸 真司、新谷 彬
名古屋第二赤十字病院医療情報管理部
- 1-2 「M言語よりのPC版SASシステムの利用」
○田久 浩志、今泉 幸雄*
東邦大学医学部病院管理研究室
*SASインスティテュートジャパン
- 1-3 「MS-Windows によるインテリジェント端末の試み」 (実演)
岡田 好一 筑波大学臨床医学系
- 1-4 「当検査センターにおけるM言語を用いた検査サブシステムの開発経過」
岩城 健一 (株)フジモトバイオメディカルラボラトリーズ
- 1-5 「汎用機とM言語システムの接続」
阿部 敏明 (株)大阪血清微生物研究所システム部

◎一般セッションII: 画面制御 【11:00~12:00】

座長: 岡田好一 (筑波大学)
: 松本重雄 (システム技研)

- 2-1 「MS-WindowsとM言語による電子カルテシステムの開発」
○嶋 芳成、小畠 興二*、谷掛 駿介**、喜多野三夫***
、岡田 怜****、笹原 茂****
日本ダイナシシステム株式会社
*小畠診療所
**谷掛整形外科診療所
***喜多野西大寺診療所
****株式会社リコー
- 2-2 「MWAPI (M言語WINDOWS-API)による電子カルテの試み」
○山下 芳範、山本 和子*、須藤 正克
福井医科大学医学情報センター
*島根医科大学医療情報学講座
- 2-3 「MS-Windowsにおける電子教科書の試作」
○永田 守秀、山下 芳範*、岡田 好一**、高橋 隆
京都大学医学部附属病院医療情報部
*福井医科大学医学情報センター
**筑波大学臨床医学系
- 2-4 「マルチウインドウ・カラーグラフィックス環境での会話型画面設計ツール
Screen maker」 (実演)
里村 洋一、山崎 俊司、○鈴木 隆弘
千葉大学医学部附属病院医療情報部

 一般演題 第2日目 【9月19日(日)】 A会場(301)

◎一般セッションⅢ： 医学におけるMUMPS 1 【9:00～9:48】

座長：田久浩志（東邦大学）
：沢田 潔（名古屋第二赤十字病院）

- 3-1 「汎用シェルと医療向けシェルの知識表現と推論機構」
○今泉 幸雄、大櫛 陽一*
SASインスティテュートジャパン
*東海大学医学部病院管理学教室
- 3-2 「自然語情報をデータベースに取込むための構造言語学の応用」
○マルシオ・ピチキ・ド・アマラル、里村 洋一
千葉大学医学部附属病院医療情報部
- 3-3 「M言語とMedical Decision Making」
○鎌江 伊三夫、笹川 紀夫、林 振健、山本 和子
島根医科大学医療情報学講座
- 3-4 「個人病院に於ける病院情報システムの活用」
ーレセプト専用機からMUMPSマシンに変更してー
○土屋 喬義、土屋 恭子、木村 一元*
土屋小児病院、獨協医科大学第一小児科
*獨協医科大学総研ME室

◎一般セッションⅣ： 医学におけるMUMPS 2 【9:48～10:36】

座長：小森 優（京都大学）
：今泉幸雄（SASインスティテュートジャパン）

- 4-1 「千葉大学病院医療情報システム(system-CHIBA)におけるシステム運用」
○本多正幸、里村洋一、鈴木隆弘、山崎俊司、高林克日己、赤井ユキコ*
、石井 晃**、中村裕義**、野口 昇**、北田光一**、清水 宗***
千葉大学医学部附属病院医療情報部
*千葉大学医学部附属病院看護部
**千葉大学医学部附属病院薬剤部
***住友電工システムズ株式会社
- 4-2 「中央検査部トータルシステムの再構築」
○中尾 満、森田 寛二、藤井 厚男、東畠 正満、翼 典之、
花坂 志郎*、平山 清和*
大阪市立大学病院中央検査部
*住友電工システムズ株式会社
- 4-3 「未熟児部および産科データベースの構築とその結合」
○田中 吾朗、渡辺 博*、木村 一元**
獨協医科大学第2小児科
*獨協医科大学産婦人科
**獨協医科大学総研ME室
- 4-4 「日本病理剖検輯報の日本語データベース」
馬場 謙介 国立埼玉病院臨床研究部

◎一般セッションV: 医学におけるMUMPS3 【10:36~11:12】

座長: 中尾 満 (大阪市立大学病院)
: 永田守秀 (京都大学病院)

- 5-1 「健診システムにおける病名と家族歴の登録サブシステムの開発」
○大櫛 陽一、笹川 紀夫*、高橋 正宏**、原 寿夫***
東海大学医学部医学情報部
*島根医科大学医療情報学講座
**郡山市健康振興財団
***郡山市医師会
- 5-2 「選択必修科目履修申告システムの開発」
○斉藤 成広、松木 三徳、菊地 良平、大櫛 陽一*
東海大学伊勢原学務課
*東海大学医学部医学情報部
- 5-3 「自然語病名の使用頻度の分布について」
○林 振健、鎌江伊三夫、山本和子、笹川紀夫、山下芳範*、須藤正克*
島根医科大学医療情報学講座
*福井医科大学医学情報センター

◎一般セッションVI: MUMPSの新技术 【13:00~14:00】

座長: 馬場謙介 (国立埼玉病院)
: 藤江 昭 (住友電工システムズ)

- 6-1 「SQLによるDSMデータベースへのアクセス」
辰己 岳欣 日本デザイン・テクノロジー・サービス株式会社西日本第一統合システム部
- 6-2 「U-MUMPSクロスシステムジャーナルによるミラーシステムの構築」
○吉村 貴由、木村 隆、伊藤 徹、吉中位知郎、藤江 昭
住友電工システムズ株式会社応用システム事業部
- 6-3 「MとTPモニターとの統合の試み」
○今井 敏雄、佐藤 真美
日本デザイン・テクノロジー・サービス株式会社西日本第一統合システム部
- 6-4 「最新DSMの特徴と今後のプラン」(実演)
佐藤 比呂志 日本デザイン・テクノロジー・サービス株式会社西日本第一統合システム部
- 6-5 「逆\$ORDER関数の効用——本院の入院診療録管理システムへの応用を例に」
○馬場 謙介、石名田洋一、鈴木 孝始*
国立埼玉病院臨床研究部
*国立埼玉病院医事課

予稿集

北米MTA大会報告 MTA-NA REPORT

山下 芳範
Y. YAMASHITA

福井医科大学 医学情報センター
CENTER OF MEDICAL INFORMATICS, FUKUI MEDICAL SCHOOL

概要

今年の北米MTA (MUG) 大会は、6月21日から25日までワシントン、D. C. で開催された。

今回がMUGからMTAに名前を変えた最初の大会であり、「M」を意識した名称や看板・内容が用意された。

初日からは、チュートリアルが中心に開始され、M言語の新装備などについてのものも含め、大盛況であった。

また、1993年のANSI規格の最終案がMDCで審議されていることもあって、と話題はこの事に関するセッションが多く用意されていた。

企業展示では、インターシステムとデータツリーの合併という衝撃的な発表とともに、新しいDT-STUDENT (教育用無償版) が配られていた。

展示・発表とも目を引いたものとして、WINDOWSの結合 (MWINDOWS-API) のデモがある。全てがこの新しい仕様を装備している訳ではないが、既にWINDOWS開発支援ツールに近いものまで作成して、WINDOWSが現実的な利用範囲であることをアピールしていた。

いずれにしても、M言語の展示発表というよりは、GUI関係の展示発表と間違ふほど、WINDOWS等の利用が進んでいる。

演題の主体も、プロダクトレビューや実例の紹介が多く、一般の発表などのセッションは少なくなる傾向のようであり、私が報告した国際セッションと比べると、技術セッションの人気には驚くばかりの状況である。

この技術セッションでの発表が、まもなく (ほとんどがMTA大会の後) 製品として出てくるものが多いことから、最新の情報源として多くの人が熱心に参加する理由でもある。

また、最終日現在のMTA-NAの参加登録数は1400人とのことであり、日本の医療情報連合大会に匹敵する参加者であり、層の厚さを感じた。

今大会では、MTA大会となったこともあり、新しいMTAのロゴが発表されている。

このロゴ・デザインのマニュアルも完成し、日本にも割当分の配布が行われた。このマニュアルは、日本MTA事務局で保管している。

この他にも、昨年に引き続き、M言語を広めるためのSTUDENTバージョン (機能限定無償版) の配布が行われとのことである。

今回は、会場においてVaflemanとセットで配布されていた。

このように、Mの普及に対してこのような努力を払っているが、日本では日本語の壁があり、これをそのまま利用できないのが残念である。

日本でもこれを活用することで、Mのアピールと拡大につなげることも可能ではないかと感じた。

MDC (M開発委員会) 報告 MDC REPORT

山下 芳範
Y. YAMASHITA

福井医科大学 医学情報センター
CENTER OF MEDICAL INFORMATICS, FUKUI MEDICAL SCHOOL

概要

昨年から今年にかけては、3回のMDCが開催された。今回は予算等の関係から、93年2月および93年6月の2回の会議に出席した。

今年は、1993年のANSI規格の最終案を決めることになる関係で、かなりの新しい仕様がMDC・TYPE-Aになっている。

主な93年仕様の項目は以下の通りである。

- SSVN (システム・ストラクチャ・バリアブル) の導入
- 国際文字仕様関連の定義
 - ^\$SYSTEM, ^\$CHARACTER, ^\$JOB,
 - ^\$GLOBAL, ^\$ROUTINE, ^\$DEVICEなど
- エラーコードの定義
- 命令の追加
 - MERGE, トランザクション関係など
- システム変数の追加
 - \$ECODE, \$TLEVEL, \$DEVICEなど
- トランザクション処理
 - TSTART, TCOMMIT, TROLLBACK, TRESTART
- OMI (OPEN MUMPS INTERCONNECT)
- M言語のWINDOWS APIの定義
- SQLの規程
- GKS利用の規程

このような項目が1993年ANSI規格に盛り込まれる見込みである。

従来の文法に比べて非常に多くの新しい機能が追加される。

特に、C言語の構造体に相当するSSVNの導入やオンライン処理では必須となりつつあるトランザクション処理などは、従来のプログラミングの考え方を大きく変えるものであり、これからのM言語に新しい方向を与えるものと思われる。

また、OMI, MWINDOWS-API, SQL, GKSなどM言語相互, M言語と他の環境との結合に関するものも、時代の流れにあったものである。

このように、1993年の仕様は1990年の仕様よりも大規模な追加変更となるとともに、国際化や他言語では実現されていた機能の標準化を含めて、これからの時代に必要な機能が取り入れられる方向である。

パソコンM言語と他言語インターフェイス

(C言語、アセンブラ言語による外部モジュールの開発)

○ 沢田 潔、 木下 元一、 浅井 廣、 岸 真司、 新谷 彬
Kiyoshi Sawada, Genichi Kinoshita, Hiroshi Asai, Shinji Kishi, Akira Shintani

名古屋第二赤十字病院 医療情報管理部
Division of Medical Information, Nagoya Daini Red Cross Hospital

はじめに

コンピュータハードウェアの資源を効率的に活用するために、ハードウェア固有のオペレーティングシステム(OS)が提供されているが、パソコンM言語の標準命令・関数を使用している限りに於いて、それらの資源の活用には限界がある。その限界を超えてOSの機能をM言語で利用する為には、他言語で実行モジュールを作成し、M言語とのインターフェイスを通してそのモジュールにアクセスする必要がある。

我々は、データベース言語としてのM言語で主プログラムを作成し、M言語では達成できない、高速数値・文字列演算、ウィンドウ画面制御、グラフィック画面制御、マウス制御、日本語FEP制御、特殊なインターフェイスボードの入出力制御等の機能をサブルーチンに実行させ、M言語の強化を計った。

今回、我々はウィンドウシステム用ビデオRAMデータの読み取り、書き込み外部モジュールを作成し、実際にアプリケーションプログラムに搭載し、その稼働に成功したので報告する。

開発環境

ハードウェア：NEC社	PC-H98 model 80
ソフトウェア：マイクロソフト社	MASM Ver6.0
マイクロソフト社	MS-C Ver6.0
住友電工社	SP-MUMPS Ver2.2、Ver3.0

開発ソフトウェア

ビデオRAMデータ読み取り、書き込みの為のソースモジュールをC言語、及びアセンブラ言語で作成し、コンパイルした(VRAM.COM)。これは本体、インターフェイス部と常駐解放処理部からなるMS-DOS実行可能形式のプログラムである。インターフェイス部は、①MS-DOS内の割り込みベクターテーブルへ割り込みベクター番号(番地)を登録するに必要なヘッダーテーブル、②SP-MUMPSの\$ZINT関数からの引き数を格納する引数パラメーターテーブル、③VRAM.COM本体からの戻り値を収納する戻り値パラメータ、等から構成される。

方法

SP-MUMPS起動前に、MSM.COM(Ver3.0ではMSM.BAT)を編集し直し"VRAM.COM"を記述する。MSM.COMを起動させるとVRAM.COM内の常駐解放部が自身を主メモリー内に常駐させる事が出来る。その後SP-MUMPSを立ち上げアプリケーションを動作させる。

SP-MUMPSでは外部モジュールを制御するために\$ZINT関数が用意されている。

割り込みベクター番号、画面読み書き位置座標、テキストおよびアトリビュートVRAM切り替えフラグ、その他のパラメータを\$ZINT関数に設定しドライバーサブルーチンを作成し、アプリケーションプログラム「名古屋第二赤十字病院救急外来受付システム」の中に組み込んだ。

結果

登録、修正プログラムの中で、地区コード・受診科・入院病棟コード・医師コード等の選択を簡易ウインドウから、カーソルを移動させることにより画面から目的とするものを一行選択し、変数の中に容易に取り込むことが出来、ユーザーインターフェイスとしての機能の充実を計ることが出来た。

考察

今回開発した機能と同様のモジュール (ZVRAM.COM) を作成し、システム技研社 CCSM に組み込み、またグラフィック画面制御ルチーン (DTMGRAF.EXE) を作成し、DataTree社 DTM98 に組み込み、各 M 言語における他言語インターフェイスについて考察した。

SP-MUMPS はソフトウェア割り込み方式を採用しているため、作成した実行モジュールを予め常駐させないか、あるいは割り込みベクタテーブルに実行モジュールの開始アドレスが登録されていない場合は致命的な OS システムエラー (In Trap Halt) を起こす。

一方 CCSM では、実行モジュールのエントリーポイントは 80H 番地に定義されており、引数の最大数は限定されている。したがって数多くの引数をモジュールに渡すことはできない。また、エントリーポイントに登録されていないモジュールを、M システム上から \$ZX 関数で呼び出すと関数なしエラーが発生するのみで、致命的なシステムエラーは発生しにくい。

また、複数の呼出規約をもつ言語とのインターフェイスを単純化するために、DTM では「仲介手続き」という概念がある。仲介部は、引数並び、外部手続き呼出、結果解釈、値返しを再構成する。「仲介手続き」を行うことで他の高級言語により作成された関数とのインターフェイスが簡便におこなえる。

また、ZCALL ヘッダに登録されていない命令、関数の呼出時には、M システム上にて命令、関数なしエラーが発生するので、致命的なシステムエラーは発生しにくい。

作成した外部モジュールは M 言語起動時には RAM に常駐させ、その終了時には解放される必要がある。DTM、CCSM では M システム管理方式をとっており、M システムが常駐、解放処理を行う。しかしながら SP-MUMPS は MS-DOS 常駐プログラム方式であるため、常駐、解放処理を別に記述する必要がある。

従来の M 言語では高速なウインドウやメニュー選択などは不得意とされていたが、初心者を始め多数のユーザーに快適な操作環境を提供出来たことは大きな意義があった。プログラム作成の上でもメニューやウインドウの記述を簡略化出来たことにより保守性が高まった。

まとめ

今日、パソコンのハードウェア、OS の機能、性能は飛躍的に進歩している。安価で高速、高機能のハードウェア、OS 上で動作するパソコン M 言語システムは我々プログラム開発者にとって有用なデータベース言語である。しかしエンドユーザーに快適な利用環境を提供するには至っていない。高度な互換性、標準性を保証する M 言語体系は、非互換、非標準のコンピュータ機能の利用を阻害している。

我々は M 言語と他言語インターフェイスの作成を試み、利用するコンピュータ機能に応じたプログラム開発を行いユーザーフレンドリーな環境を構築した。

謝辞

今回の発表において、DTM98、DTM ZCALL 開発キットをご提供いただきました、日本ダイナシステム 嶋 芳成氏と鈴木 利明氏、SP-MUMPS 技術資料をご提供いただきました、住友電工社 武田 芳郎氏に深謝いたします。

M言語よりのPC版SASシステムの利用

○田久 浩志^{*}、今泉幸雄^{**}

^{*}東邦大学医学部 病院管理学研究室

^{**}SASインスティテュートジャパン

M Language and PC-SAS

^{*}TAKYU Hiroshi Department of Hospital Administration, School of Medicine, Toho University, 5-12-16, Omori-nishi, Ota-ku, Tokyo 143

^{**}IMAIZUMI Yukio SAS Institute Japan Ltd.

Nichirei Akashicho Bldg. 6-4 Akashi-cho Chuo-ku Tokyo 104

要旨

DTMで作成したデータベースを用い、PC版SASシステムでデータ解析をする場合の有効性についての評価をした。そのため、M言語で検索したデータをシーケンシャルファイルに出力した後 SAS でデータを読み込む場合と、SASデータセットでデータを検索する場合の処理時間を比較した。1種類のファイルを用いる場合、2種類のファイルを結合する場合には、データ件数によってはM言語とPC版SASシステムを使用する方が処理時間が早かった。これらの事より、数万件のデータより一部を検索して解析する場合、ディスク容量に制限がありソートがしにくい場合にM言語とPC版SASシステムを用いると有効であると考えた。

Key Words: PC-SAS, Data Tree Mumps

1. 目的

今回、福岡県の医療圏設定を評価する試み（患者数約40万人、医療施設数約4千）を行うにあたり、M言語(Data Tree Mumps)からSAS(PC版SAS)を用いる環境を、解析の補助手段として検討することになった。SASと多くのデータベース間にはSAS/ACCESS等のインターフェースをとる製品があるが、M言語環境とのインターフェースは存在しない。本論文では、IBM PS55上でPC版SASとM言語を使用した時の、データの検索処理速度の評価について報告する。

2. M言語とSASのインターフェース手法

M言語（以下Mで示す）とSASとを利用する方法は幾つか考えられる。以下にMとSASとのインターフェースをとる方法について検討する。

A. SAS側でSAS/ACCESSを使用し、M側でSQLを実行する製品を使用する方法

既に、Mの上ではM/SQL等のSQLを解釈する製品がある¹⁾。この手法は、SAS、Mの両方で新規のプログラム開発の必要はないが、新規のソフトウェアの購入が必要となる。

B. SASでOMI(Open MUMPS Interconnect)を用いる方法

異機種、異装備のMUMPSアプリケーション間で、リモートデータベースを実現するためのネットワークプロトコル, Open Mumps InterconnectをSAS側で実現する手法である。M側よりSAS側にデータを転送してデータセットを作成し、その後に処理を行う考えである。この場合、SASでの新規プロダクトの開発が必要となる。

C. インターフェース用プログラムを開発する方法

アセンブラ、C等でM側のグローバル変数を取り込むインターフェースプログラムを記述する方法である。既にAccess File Description, Master File Descriptionの2つの定義ファイルを用いてFOCUSとDSMの間のインターフェースにこの手法が採用されている^{2,3)}。処理速度の上昇は期待できるが、M言語の動くプラットフォーム毎の新規プロダクトの開発が必要となる。

D. シーケンシャルファイルを経由する方法

M側のプログラムでSASのプログラムとデータを作成した後、一度シーケンシャルファイルに出力し、その後、SASを起動してプログラムを実行する方法である。どのM言語の動く環境でも実現可能であり、SAS側で新規の開発が不必要である。その反面、M言語の終了時間、SASでのデータの読み込みまでのオーバーヘッドが大きい短所がある。今回は、PC版SASシステムを用いることより、Dの外部ファイルを経由する方法を採用した。MでSASのプログラムを生成するプログラムは別途作成した。

3. M側でのSASプログラムの生成

SAS側のPROCステップをデータ解析エンジンとして使用することを考慮し、まず第一段階としてSASにおけるCHART, PLOT, PRINT, FREQ, CORR, MEANS, SUMMARY, UNIVARIATE, TABULATEの各PROCステップを利用することを計画した。そのためには、M側よりSASのプログラムを生成するルーチンを記述する必要がある。しかし、SASランゲージを使用したプログラムは、非常に奥が深いものがあるので、第一段階では最小限の機能を用いれば良いと考え、もし必要があれば生成されたプログラムを修正して使用すると考えた。これらを考慮して、プログラムとして`%UUsasを作成した。これはローカル変数に必要なSASランゲージを文字列として代入して、対応するプロシジャ名のルーチンをよべば、必要なSASプログラムを生成するものである。一方、PROCステップと異なり、外部ファイルを入力して、SASデータセットを作成するプログラムでは、読み込む変数の順番を指定しなければならない。そこで、データ入力プログラムの生成に関しては、ローカル変数に、読み込む変数の順番に対応して、変数名、FORMATに関するSASステートメントを代入するものとした。また、SAS側にわたすデータはカンマで区切った形式とし、SAS側でINFILE指定により読み込むものとした。

例：

```
M側 S A="MYDIR.BUF", A(1)="ID", A(2)="NAME $", A(3)="CLASS"
     S X=$$data~%UUsas(. A)
```

```
SAS側 data MYDIR.BUF;
      infile 'out.dat' dlm="," missover ;
      input ID    NAME $    CLASS
      ;
run;
```

4. 処理速度の比較

SASとMで、データを検索して新規のデータセットを作成するまでの速度を検討するため、福岡県の京筑地方の医療圏調査で使用した患者8460人と医療機関99施設のデータを用いた。患者のデータは、施設番号、患者番号等の55変数よりなる。1データあたりの容量は平均132byteである。医療機関のデータは施設番号、開設者の種類、標榜科目等全体で14変数、116byteよりなる。事前に施設毎の患者データは、100, 200, 500, 1000, 2 000, 4660件に設定した。このファイルを2回使用し、16920件のデータも用意した。

使用した機械はIBM PS55/NOTE N23x, Memory 6 Mbye, Hard Disk 80 Mbyteである。DOS/V上にQEMM-386とTRを搭載し日本語の使用できる環境としてPC版SASのVer. 6. 04を使用した。M言語はData Tree Mumps (以下DTM) Ver. 4. 3を用いた。

残念ながら、現段階ではDTMの内部よりSASのプログラム起動する手立てはない。そこで、DTMとSASを用いる処理時間の測定には、以下の手順でDTMでの検索とSASのプログラム実行を連続して行うバッチファイルを作成し処理時間を求めた。

```
DTMの起動   グローバル変数のデータ検索プログラムを実行
              結果をシーケンシャルファイルへ出力
              SASで用いるデータ読み込みプログラムの生成
SASの起動   データ読み込みプログラムをバッチで実行
```

次にSASのみで行う処理時間の測定は、M側のグローバル変数と同じデータをSASのデータセットで作成しておき、それからIF文で該当するデータのみを検索し、新規データセットを作成する場合の時間を求め、上記の時間と比較した。

4.1 1種類のファイルでの処理速度の測定

患者データのみを用いて、データ検索を行う場合の検索時間を比較した。患者データは条件1-4は8460件、条件5-6は16920件である。

条件1	SASデータセット上でn件検索し、新規のSASデータセットを作成する。
条件2	グローバル変数上でn件検索し、一度シーケンシャルファイルへ出力。 それをSASで読み込み、SASデータセットを作成する。
条件3	SASデータセットよりn件検索し3変数で、新規データセットを作成する。

- 条件4 グローバル変数上でn件検索し、3変数をシーケンシャルファイルに出力。
それをSASで読み込み、SASデータセットを作成する。
- 条件5 条件3に同じ、但しデータ件数は16920件
- 条件6 条件4に同じ、但しデータ件数は16920件

図1に各条件の比較を示す。8460件を対象とする場合、変数全てを用いる場合、検索結果の件数が1200件程度までは、DTM側で検索したほうが処理時間は早かった。実際の多重分割表を作成する場合、変数は3, 4個の場合が多い。これに相当する条件3, 4を比較すると4600件のデータ検索までは、DTM側で検索したほうが処理時間は早かった。条件3と5を比較すると、SASのみでデータ処理を行う場合、オブザベーションの数に比例して処理時間が増加した。条件4と6とを比較すると処理時間はほとんど変わらなかった。これはグローバル変数の検索の場合、キーの数に比例して処理時間が増加し、データ数全体とは関係しないためである。

4.2 2種類のファイルを横方向に結合する場合の処理速度

患者のデータに、その施設番号をキーとして横方向に医療機関のファイルを結合する場合の処理速度を比較した。結合するデータは、キーの他に住所、施設種類の2変数である。SASデータセットのみで処理を行う場合、一度患者データを施設番号でソートしてから医療機関のデータをマージをする必要がある。そのため、事前に医療機関のデータセットは施設番号でソートを行った。DTM側では、まず患者データより施設番号を求め、これをキーとして医療機関のグローバル変数を求め、患者データに必要な変数を結合した。処理の条件を以下に示す。患者データは条件7, 8は8460件、条件9, 10は16920件である。

- 条件7 SASデータセットよりn件検索し、3変数で新規のデータセットを作成。
それを施設番号でソートの後、施設番号をキーとして、医療機関データより住所、施設種類の2変数をマージする。
- 条件8 グローバル変数よりデータをn件検索し3変数を得る。
施設番号をキーとして医療機関データより住所、施設種類の2変数を結合し一度シーケンシャルファイルに出力。そのファイルをSAS側で読み込み、SASデータセットを作成する。
- 条件9 条件7に同じ、但しデータ件数は16920件
- 条件10 条件8に同じ、但しデータ件数は16920件

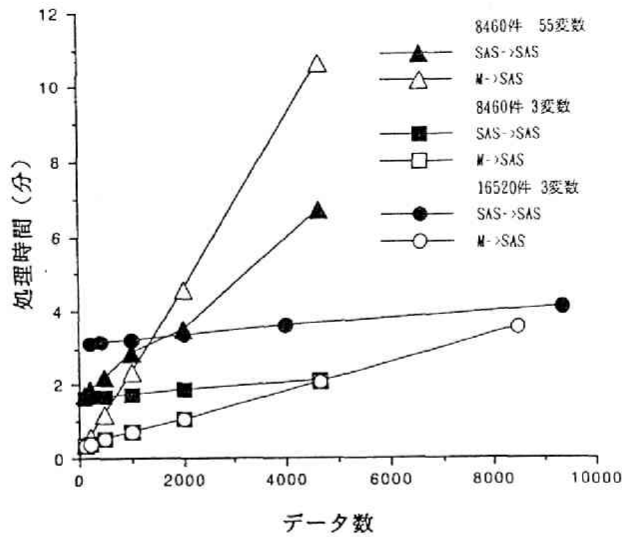


図1 1種類のファイルの処理時間

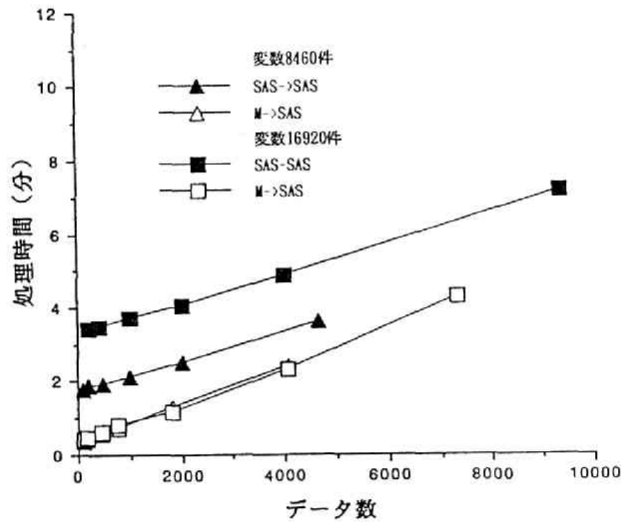


図2 2種類のファイルの結合処理時間

図2に各条件の処理時間の比較を示す。条件8と条件10で処理時間は同じである。どの条件でもデータ処理時間は、DTM側で一度検索したほうが早かった。

5. 考察

今回の結果より、単一のファイルの使用では55変数全てを使用する場合、検索結果の件数が1200件程度までは、DTMで検索したほうが処理時間は短かった。使用する変数を3個に限ると4600件のデータ検索まではDTMで検索したほうが処理時間は短かっ

た。また2種類のファイルを結合する場合、8460,16920件のデータを対象とする場合はDTMで検索したほうが処理時間が短かった。これは、グローバル変数の構造とSASのデータセットの構造に起因するものである。SASデータセットはインデックスとファイル部分より構成され、その検索時間はデータ全体の数に比例してしまう。一方、複数のファイルを結合する場合、SASではソートとマージが必要となる。医療圏の調査では、施設数4000、患者数400000程度の大量データを扱う場合が頻繁に生じる。この事はディスクやメモリーの資源を豊富に使用できるVAX, UNIXなどの計算機環境では問題とならないが、PC版SASの動く環境ではかなり問題となる。

DTM, PC-SASの起動終了のオーバーヘッド、シーケンシャルファイルへのデータの入出力時間は、少量のデータではデータ処理効率に大きな影響を与える。しかし、大量のデータを扱う場合では、相対的に影響がすくなくなる。これらの事より、M言語とPC-SASを使用するような環境では、データの数的大量にある、使用する変数が少数で検索する対象が全体の一部の場合など、M言語の環境よりPC-SASを利用する価値はあると言えよう。また、VAX、UNIXなどの環境ではデータをM側からSAS側に渡す方法も、メールボックス経由などの高速な手段も考えられる。今後M側の資源を解析するのに、SASをデータ解析エンジンとして使用するのも有意義と考える。

6. まとめ

従来、M側でデータ処理を行う場合、その都度ルーチンを記述する必要があった。そのため、病院等で戦略情報システムとしてMのデータベースを利用しても、重要な情報を迅速に提示するのは難しい問題であり、誰でもが可能というわけではなかった。医療圏評価の解析をするにあたり、複雑な表を処理できるTABULATE、多重分割表を生成するFREQなどは非常に有用であった。今後は、MとSASの長所を活かして使用することが重要である。そのため、次の段階ではMでは不得手であるグラフィック機能を利用するため、SAS/GRAPHの利用を考えている。

参考文献

- 1) Jared S. Corman and Daniel Pasco: SQL and MUMPS, MUMPS Computing, Vol 22, No. 1, P. 24-27, 1992
- 2) Sebastian Holst: Integrating Digital Standard MUMPS (DSM) and the FOCUS 4GL, MUMPS Computing, Vol 22, No. 1, P. 29-31, 1992
- 3) Focus for VAX/VMS, Interface to Digital Standard MUMPS Release 6.1, FOCUS Information Builders, Inc. 1991

E-Mail Address	Nifty-Serve Internet	MAF00072 taq@tansei. cc. u- tokyo. ac. jp
----------------	-------------------------	--

○ 岡田好一

筑波大学 臨床医学系

1、はじめに

最近、30万円程度でウィンドウが使える高機能なパーソナルコンピュータが手に入るようになった。本来ならば、ネットワーク対応でWindowsに対応したM言語を使うのが、利用者指向のデータベース環境としては、良いと思われる。しかし、C++言語の開発環境は非常に安価で優秀なので、ここではあえて、M言語はデータベースサーバーとしてのみ使い、C++言語を利用したWindowsプログラムを利用者インターフェースに使用してみた。

2、材料・方法

富士通のワークステーションS-4/IX(Sun4/IPX)に、住友電工のU-MUMPSで簡単な応答ルーチンを走らせておく。パソコン(IBM PS/V 2450 WV8, CPU 486SX-20MHz, RAM 8MB, VRAM 1MB)ではポーランドC++で書かれたマイクロソフトWindowsのプログラムが利用者とのインターフェースを行う。利用者がパソコンを操作すると、パソコンのプログラムがRS-232C経由でワークステーションに対しデータを送り、ワークステーションが応答し、結果をパソコンのディスプレイに表示する。

ワークステーションのM言語はデータベースに対する基本的な応答のみを受け持つ。M言語自身のインタプリタをそのまま用いてもよかったが、エラーチェックと再送信をさせるため、独自のプロトコルを作った。応答ルーチンは、set(read/write), kill, lock, \$data, \$order, \$queryなどの基本的な問い合わせ・書き込みをするのみである。データベースのスキーマのチェックはしない。ただし、添字に適したコードのチェック、文字列の長さのチェックは行う。

パソコンではC++のみを用いてウィンドウプログラムを書く。ワークステーションのM言語は簡単な応答しかしないので、データベース操作のプログラムはパソコン側で記述することになる。

3、考察

本稿では、専用の通信路を持ち、サーバー、端末双方にディスクと計算能力がある上記のようなシステムを、インテリジェント端末システムと呼ぶ。MS-Windowsは疑似的ながらマルチタスクシステムであるので、利用者からみれば、サーバーを利用したワークステーションと言ってもよいであろう。

問題は、Windowsのプログラミングである。MS-Windowsは疑似的なマルチタスクシステムであり、通常のCあるいはC++言語の感覚ではプログラムできない。そのようなウィンドウプログラミングで、従来のM言語によるデータベース操作と同等のことができるかどうか、本稿の目的の一つである。

目的の第二は、サーバーとパソコンの役割分担である。今回はM言語側ではデータベースのスキーマのチェックなどはしていない。M言語は単にワークステーション上での添字付ファイルを提供するのみである。これが本来のM言語の考え方に最も近い利用法であろう。

別の考えとして、スキーマの管理はM言語にさせ、通信プロトコルの上位階層はデータベースの問い合わせ言語としてもよい。データベースサーバーというからには、スキーマのチェック機能は不可欠であろう。この場合は、問い合わせ言語はM言語外となり、新たな設計が必要となる。

むろん、最初に述べたように、ネットワーク対応でウィンドウズ対応のM言語があれば問題は解決する。それには2つの方法があって、1つはM言語にウィンドウの操作を直接させることであり、もう1つはマルチタスク環境を利用して、M言語が(ウィンドウプログラムとの)プロセス間通信をサポートすることである。おそらく両者とも現状で手に入る技術であろう。

しかし、C++のウィンドウ開発環境は良好で、安価であり、現状ではウィンドウズシステムのバージョンアップの際にも有利ではないだろうか。このような側面からみれば、インテリジェント端末の考え方にも、評価の余地があると思う。

汎用機とM言語システムの接続

(株) 大阪血清微生物研究所 システム部 安部敏明

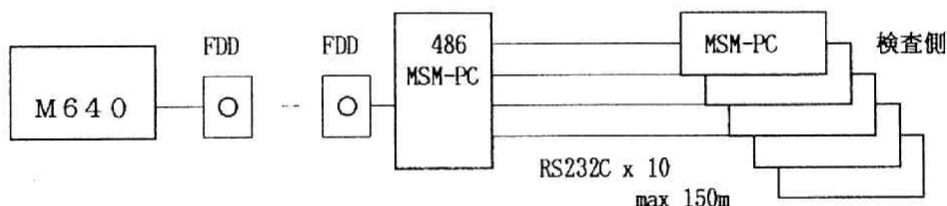
日立製作所製M640汎用機と、M言語を用いた検査システムの情報交換にLANを利用し良好な結果が得られたので報告します。

システムの全体的な作業を汎用機で処理し、実際の検体検査を行う検査システムはM言語を利用しPC486機を多用して処理を行っている。

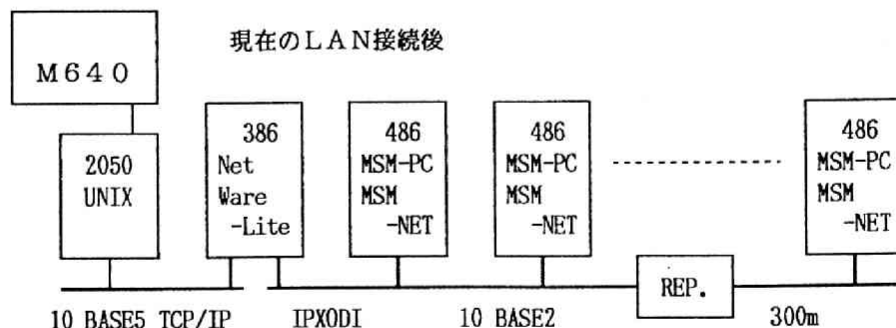
従来は、汎用機と486機間の情報交換には、一旦フロッピーディスクに交換情報を落とし、それを486機で読み込んだ後にRS232C(9600bps)を利用して情報を検査の486機に分配し、検査側の486機の情報も逆のルートで汎用機に転送していたため、相互の情報転送に時間がかかりすぎ、又、バッチ処理として対処していた。

システム全体として、この間の情報交換がシステムのボトルネックとなっていた。

LAN接続前の状態



現在では汎用機と検査側486機間の情報交換をスムーズに運営でき、ボトルネックを解消するために、RS232Cの利用を廃止してLANを利用した接続に変更した。



ハード、ソフト環境と方法

- 1 M640のLAN接続インターフェイスとして、2050（日立製UNIX機）を利用し汎用機のLANの接続口を確保した。
尚、M640と2050間は、HIFIT（ファイル転送ユーティリティ）を用いた情報交換を行っている。
- 2 386機にはNET-OSであるNetware Lite（英語版）と、NICを2枚（EXOS300T,SMC3016）を登載し、汎用機と486機間の情報受渡しのためのファイルを作成させた。。
- 3 各486機には、従来のMSM-PC上にMSM-NETを登載し、10BASE2のケーブルを引いて386機に接続を行った。
- 4 2050では、C言語を用いてHIFITで作成されたファイルをMS-DOS形式のファイルに変換し、又、この逆の処理を行う。
- 5 386機からRCPコマンドを出して2050側のMS-DOS形式のファイルをLAN経由で読み書きしている。
- 6 同様に486機側は、M言語により直接386機のNetware上のファイルをLAN経由で読み書きを行わせている。

本来、386機とNetwareを用いなくても可能ではあるが、2050側のC言語でのソフト開発の工数削減を目的とできるだけ簡易に汎用機とのLAN接続をするために、敢えて、386機とNetwareを中間に採用した。

多少の問題点もあるが、実用上差障りがなく運営できボトルネックも解消された。

MS-WindowsとM言語による電子カルテシステムの開発
Development of Electronic Medical Record System using MS-Windows and M Language

○嶋 芳成、小島 興二*、谷掛 駿介**、喜多野 三夫***

岡田 怜****、笹原 茂*****

Shima Yoshinari, Kojima Kyoji*, Tanigake Shunsuke**, Kitano Mitsuo***

Okada Satoru****, Sasahara Shigeru*****

日本ダイナシステム株式会社、*小島診療所、**谷掛整形外科診療所、
喜多野西大寺診療所、*株式会社リコー

はじめに

我々は、すべての診療情報をコンピュータに登録し、従来の紙に手書きするカルテの代わりに利用するシステムを、電子カルテ(または電子化カルテ)と呼んでいる。このようなシステムの効用は明らかであるが、実際にはコンピュータへのデータ入力作業が煩雑であるため、現場の医師に受け入れられず、完全に従来のカルテの代用として使われているシステムはほとんどないのが現状である。

そのなかで、共同演者の一人である小島¹⁾²⁾は1989年から、内科の無床診療所で、紙の診療録を用いず、電子カルテシステムのみを用いて診療を行ってきた。

3年余り利用した現在、システム上いくつかの問題点が明らかになってきた。このシステムはCOBOLベースで開発され、全角文字と半角文字が混在できないなど文章データの扱いに若干制限があること、画面がかならずしも理想的なものではないこと、患者の問題リストから問題を選択してそれに関する病歴を入力するという方式がしばしば困難な作業となること、医事会計情報をもとにデータベース化しているために病歴としては不要なデータがあること、などである。

新システムの開発

演者らは、このような問題点を解決し、システムを改良し、より広く利用できるようにするため、小島のシステムをもとに、電子カルテを再開発することにした。

ハードウェアとしては、最近低価格化が著しく、また性能も飛躍的に向上したパソコン(PS/4-(株)リコー)によるネットワークシステムを用いることとした。また操作のしやすいこと、市販パッケージソフトが利用できること、近い将来、安価に画像処理やマルチメディア機能を付加する可能性を考え、グラフィカルユーザーインターフェース(GUI)として代表的なMS-Windows(マイクロソフト(株))をベースとし、アプリケーションプログラム開発には、M言語システムであるDTM(日本ダイナシステム(株))とそのWindows APIであるDT-Windowsを採用した。

システム設計の基本方針

このシステムは、無床診療所を対象とし、医師自らがデータを入力することを前提としている。注射、処置、検査などの指示はコンピュータを介すよりも、メモ等で指示した方が簡単で迅速なので、処方以外の指示入力機能は省略した。

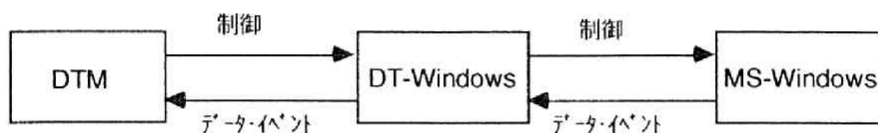
診療所の実際の業務では、診療録の記載は医事会計業務と不可分である。会計に必要な情報も医師が入力し、それを診療情報として登録するとともに、その場で診療費の計算をすることができることとした。医師が診療費について患者と相談しながら診療方針を決定することは、インフォームド・コンセントという視点からも意義がある。

家族歴などの背景情報や所見などの文章情報は、ワープロ方式でキーボード入力するものとした。文章情報を入力してから、文章(の一部)を選択し、次にそれがどのような種類の情報であるか、どのような問題に関連するかを指定することとした。この点が、問題点を選択してから記載する方式であった従来のシステムと大きく異なる点である。この結果、この電子カルテを、問題指向システム(pos)的に利用することも、あるいはすべてを無構造な文章データとして扱うことも可能となっている。

マルチウィンドウやボタン、プルダウンメニューなど、MS-WindowsのGUI機能を用い、文章入力以外はキーボードを使わなくても操作できるようにしている。画像、音声、動画などについては、今回の開発には含まなかった。

M言語のWindow APIについて

DT-Windowsは、DTMとMS-Windowsを連係するためのAPI(アプリケーションプログラムインターフェース)である。DTMからは、`write /wxxx` という命令でMS-Windowsの機能をコントロールしたり、MS-Windowsからイベントを取得することができる。



一方、MUMPS開発委員会(ANSI X11委員会)ではM Windowing API(MWAPI)³⁾というウィンドウ環境でのM言語の標準仕様を策定している。これは、MS-Windows、X-Windowsなどベースとなるウィンドウシステムによらず同一のアプリケーションを稼働することのできる移送性の優れた規格であること、イベントループをM言語のアプリケーションが管理する必要がないなどの特長をもつ。残念ながら、1993年6月の時点ではDT-Windowsがこれに対応していない。今回のシステムではウィンドウ定義方法などをMWAPIに準じた形にすることで、近い将来これに容易に移行できるように工夫している。

開発の現況

このシステムは、1994年4月稼働開始を目標に現在開発を進めている段階である。

考察

医師による指示入力を採用した病院情報システムは、現場の医師には決して評判の良いものではない。これはコンピュータ化が中途半端で、紙の診療録と併用しなくてはならないことも大きな理由であろう。紙の診療録よりも便利なシステムであれば、医師は喜んで利用するであろうし、そうなればコンピュータの大きな効用が期待できる。

コンピュータシステムを、医師のみならず現場のユーザーにとって、より分かりやすく、より便利にするには、近年発展しているGUIは有力な技術である。M言語アプリケーションもそのような方向へ発展すべきであることは間違いない。そのためにはMWAPIは強力なツールとなるだろう。

文献

- 1) 小島興二、他：電子カルテシステム - 日常診療における使用 - 日本病院会雑誌、Vol.37、No.9、1990
- 2) 小島興二、他：電子カルテの臨床使用例 - 診療録管理、Vol.3、No.1、1991
- 3) Gardner, G., et al: M Windowing API, MUMPS Development Committee, ANSI/X11/SC11/TG4/93-3, 1993

M W A P I (M 言 語 W I N D O W S - A P I) による電子カルテの試み

A TRIAL FOR ELECTRONIC MEDICAL RECORD USING MWAPI

○山下 芳範、山本 和子*、須藤 正克
Y.YAMASHITA, K.YAMAMOTO*, M.SUDO

福井医科大学 医学情報センター
*島根医科大学 医療情報学講座

FUKUI MEDICAL SCHOOL, CENTER OF MEDICAL INFORMATICS
*SHIMANE MEDICAL SCHOOL, DEPT. OF MEDICAL INFORMATICS

はじめに

本学の情報システムでは、開院時より診療病歴データベースの構築を行い情報の蓄積を行ってきた。オーダーの導入に伴い、蓄積しているデータベースの内容も増加している。本学では、このようなデータベースを基に、診療録に相当するデータベースの構築を進めている。これは、オーダーと診療録作成という重複作業を軽減することも目的としている。

本学では、このMWAPIが作成される以前より、M言語とWINDOWSの結合を行い、試作を進めてきた。

最近では、M言語の規格も大きく変化しており、WINDOWSに対するハンドリング(MWAPI: M WINDOWING API)についても装備されつつある。

そこで、この規格に対応すべく従来のシステムを、新しいWINDOWSハンドリングの方式を利用することとなった。

MWAPIを利用すれば将来にわたって資産として活用できるだけでなく、このようなシステムの標準化につながることへの価値は大きい。

概要

今回の試作は、新しい装備であることもあって、試験的なバージョンで行っている。しかし、従来の独自のインターフェースと違い、将来的に利用できることが保障される方法であるので、このMWAPIに合わせることの意義は大きい。現時点では、完全なインプリメントとなっていないが、従来の方法を十分に移行することができる。

従来の方法とMWAPIとの大きな違いは、イベントの取扱いである。

従来はイベントポーリングとして、WINDOWSとM言語のインターフェース部分でイベントの受付とスタックを行いM言語から1つずつ取り出していたが、MWAPIでは各イベントから直接ルーチンへの起動というイベント・ドリブンとなった。(図1、2)

福井医科大学 医学情報センター 福井県吉田郡松岡町下合月23-3
FUKUI MEDICAL SCHOOL, CENTER OF MEDICAL INFORMATICS
23-3, SHIMOAIZUKI, MATSUOKA, YOSHIDA, FUKUI
0776-61-3111 EX.3580 E-MAIL: yyama@fms101.fukui-med.ac.jp

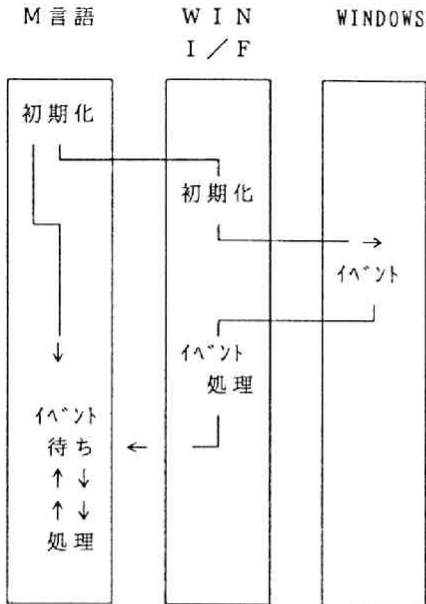


図1. 従来のハンドリング

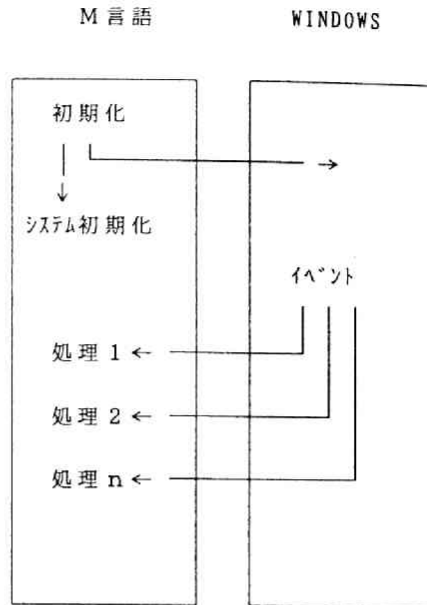


図2. MWAPlでのハンドリング

このように、従来の方法と異なる点は、初期化およびイベントのハンドリングのためのメインルーチンとそれぞれの処理の受け渡しである。

また、WINDOWS上のボタンやリストボックスなどの要素の定義もできるようになったので、画面設計などで柔軟な対応が可能となっているが今回は用いていない。

あくまでもルーチン上のコーディング方法が変更の対象となっているので、グローバルに関しては従来の物をそのまま利用している。

まとめ

M言語にMWAPlが装備されることは、このようなグラフィカルなインターフェースを用いても移送することができるようになり、病院システムなどの大規模なアプリケーションでは資産の継承・保護が期待できる意義は大きい。

しかし、従来からのM言語の考え方とは大きく違い、イベント・ドリブンの要素が入ってくることや新しい命令・変数の利用が必要など、プログラミングのスタイルが大きく変わる。少なくとも、WINDOWS上のソフトをC言語で記述するのに等しい考え方が必要と思われる。

このようなことから、従来からのM言語プログラマーに利用させるには、GUI設計支援のユーティリティーの必要性を感じた。

参考文献

MDC 93年6月 PRE-MEETING MAILING DOCUMENT

MS-Windowsにおける電子教科書の試作 Development of Electronic Text Book for MS-Windows

○永田守秀、山下芳範*、岡田好一**、高橋 隆
○M.Nagata, Y.Yamashita, Y.Okada, T.Takahashi

京都大学医学部附属病院 医療情報部
福井医科大学 医学情報センター*
筑波大学 臨床医学系**

Department of Biomedical Informatics, Kyoto Univ. Hospital
Fukui Medical School
Tsukuba University

概要

MS-Windows上でMumpsを使い、医学電子教科書を試作した。
診療支援のための電子教科書を目指すなら、電子カルテなどのアプリケーションと連動して動作する必要がある。これを実現させるための環境として、パソコンのMS-Windowsを利用する。Mumpsの不得意とするユーザーインターフェース部分はVisual BASICで設計し、MumpsとはDDEにより情報の交換をおこなっている。

Abstract

We are developing an medical assistance electronic textbook on MS-Windows by using Mumps. It is necessary that this textbook is used with other applications(ex. electronic medical record). By using MS-Windows, the multi application platform is becomes available. GUI of the electronic textbook is designed by Visual BASIC and it connected with Mumps through the Dynamic Data Exchange.

Keyword

電子教科書、MS-Windows、DTM-PC

1 はじめに

教科書のデータを電子化し活用する際、文章データだけでなく可変長のデータとすることによりMumpsで容易に処理することができる。しかし、パソコンのMumpsで写真・図・グラフなどのイメージデータを扱う場合、これまでは一般的な表示方法がなく専用のグラフィックドライバを開発する必要があった。このため、我々が今まで開発してきた電子教科書は、特定の機種（富士通FM-Towns）に依存したシステムとならざるを得なかった。さらにこの電子教科書は、文章の検索・参照の画面と画像の表示の画面とは画面モードが切り替わるため、文章と写真の同時表示ができないうえに、文章画面はキーボードだけ、画像画面はマウスだけの操作となり、ユーザーにとって操作性に一貫性のないものとなっていた。

しかし、パソコンのMS-Windowsの普及とMS-Windows上のMumps(DTM-PC)の登場により、データベース操作をMumpsで行い、ユーザーインターフェースの設計をMS-WindowsのGUIに容易にゆだねられる環境が揃いつつある。これにより文章と画像の混在した画面が構成でき、また診療支援のための電子教科書として活用できるようになった。

2 システムの概要

MS-Windows版の電子教科書は、以前つくられたFM-Towns版からの移植であり本質的な違いはない。これまでのSP-Mumpsでの処理をDTM-PCに置き換え、C言語で開発されたグラフィックドライバが行っていた処理を、Visual BASICで置き換えGUIの設計を行った。

Visual BASICを採用することで、画面に絵を張り付ける要領で簡単に画面設計ができ、試行錯誤をしながらでも変更が容易に行うことができた。Visual BASICとMumps間は、DDEにより情報の交換を行う。これにより表面上はVisual BASICの画面を操作していてもバックグラウンドでは常にMumpsが待機していることになる。

GUIと連動させる方法として、これとは別に、Mumpsから直接Window関数を操作する方法がある。しかしMumpsでGUIプログラムをコーディングするよりも、独立したアプリケーションとして設計したほうが良い。つまり、基本的な情報の受け渡し規則だけ決めておきMumpsではDB操作のみを行い、データの入出力は別のアプリケーションが独自に行うようにする。これにより電子カルテなど他のMS-Windows上のアプリケーションとも、直接情報を交換することが可能となる。

画像データは1画像1ファイルとし、Mumpsからはファイル名を指定することにより表示を行う。DOSの階層ディレクトリを利用するこの画像ファイル管理方法はMumpsの管理外にあるデータとみなされるので、インテグリティが保証できない。できれば画像データもグローバルとして扱うようにしたいのだが今後の課題として残されている。画像データはBMPファイルとして扱っている。これはVisual BASICの制約によるためである。

教科書データは、(株)南江堂発行 吉田修監修「ベッドサイド泌尿器」からテキストデータと画像データの一部を扱った。

3 まとめ

パソコンのMS-Windows上で動作する電子教科書を試作した。

GUIの向上により操作性が以前に比べ一貫性をもつようになり、またMS-Windowsのアプリケーションとすることで、他のアプリケーションとの同時活用が可能となった。今後はシステムの一部として実用性を高める改良を行っていく。

マルチウインドウ・カラーグラフィックス環境での会話型画面設計ツール Screen maker

里村 洋一、 山崎 俊司、 ○鈴木 隆弘
千葉大学医学部附属病院 医療情報部

SATOMURA Yoichi, YAMAZAKI Synji, SUZUKI Takahiro
Division of Medical Informatics, Chiba University Hospital, Chiba

1. 開発の目的 : M言語はVDT画面の操作に関して、何らのコマンドや関数を準備していない。これは、VDTの機能に依存するこれらに標準的な仕様を与えることが困難であり、言語上の規定がプログラムの移送性を阻害すると考えてのことである。しかしながら、オンラインアプリケーションが主流となった今日、ユーザーは、マッキントッシュに似たGUIを要求し、これに応えるための画面設計がルーチン作成過程の大半を占めるまでになってきた。パーソナルコンピュータを端末として利用する場合、パソコンが備えているカラーグラフィックスやマルチウインドウの機能を利用するのが良いが、それとても、M言語で直接操作できるものでもない。この問題を解決するため、われわれは、住友電工のMWE80 (VT80準拠の Multi-Window Emulator) をベースとして、会話型でマウスを使いながら、画面設計を行ない、入出力画面を設計し、同時にルーチンを自動作成するツールを作成した。

2. 機能の仕様 : このツールに要求され、実現された仕様は次のようである。
完成されたルーチンの備える機能

- 1 様々なサイズのウインドウを多重に開けること。
- 2 ウインドウの中では、マウス、カーソルキーで入力位置を移動指定できること。
- 3 各入力項目毎にチェックルーチンを設定できること。
- 4 入力補助画面やヘルプ画面が入力項目に応じて別ウインドウに表示できること
- 5 25行目にファンクションキーが設定できること。
- 6 ウインドウの中にボタンを設定でき、これに機能を持たせること。
- 7 入力項目の他にデータ表示が可能で、これが、入力項目と対応付けが可能なこと。
- 8 表示文字のカラーやモードを自由に選べること
- 9 入力位置での文字列編集が可能なこと。

ルーチン作成時に使える機能

- 1 表示文字の場所やサイズ、色等の指定は全てマウスで行なう。
- 2 表示文字は、指定した画面のそのままの位置で入力できる
- 3 設計時に必要な操作ガイドは25行目に表示される。
- 4 指定を中断した場合、それまでの状態をセーブし、同じ状態から再開が可能。
- 5 順次、直前の項目指定を取り消して、やり直しが可能。
- 6 一旦完成したものも、修正プログラムによって、同様の操作で修正が可能。
- 7 画面設計が完了したら、直ちにそのルーチンの実行とデバッグへの移行が可能。

3 システムの構成 : 次の3つのツールが用意されている。

- 1 アプリケーション作成ツール 上記の機能をそなえた作成ツール (SCREENMK)
- 2 アプリケーション修正ツール 任意の項目を選んで修正、削除、移動を行なう修正ツール (SCREENSY)
- 3 アプリケーション実行エンジン 作成されたアプリケーションを実行する共通のルーチン (XCURSER7)

また、このツールによって作成されたアプリケーションは次の5つのルーチン及び画面パラメータのためのグローバル変数より構成される。両者とも画面設計終了時に自動的にディスクにセーブされる。

- 1 メインルーチン (^*****m) : 各種画面パラメータの設定、XCURSER7の呼び出し、データの表示(別ウィンドウを使う)などを実行する。
- 2 ヘルプルーチン (^*****h) : 入力項目に関するヘルプウィンドウの表示、選択項目リストからの入力などを実行するサブルーチン集合
- 3 チェックルーチン (^*****c) : 各項目の入力値をチェックするサブルーチン集合
4. ファンクションキー ルーチン (^*****p) : 12のファンクションキーの機能を指定し実行するルーチン
- 5 ボタンルーチン (^*****b) : ウィンドウ中に設定されているボタンの機能を実行するサブルーチン集合
- 6 パラメータグローバル (^SCREENMK) : 表示及び入力項目の各種パラメータ及び属性を記録したグローバルで、アプリケーション実行時に呼び出されて、ローカル変数にセットされる。

これらの相互関係をFig.1に示す。また、作成されたルーチン群の実行過程をFig.2に示した。

4. プログラムの詳細

4.1 入力項目 : 入力項目は項目名、文字色、その表示位置、入力領域の位置とサイズなどが、マウスを使って定義できる。このとき、項目番号が自動的に与えられる。以後、修正およびアプリケーションは、この項目番号をキーとして実行される。

4.2 データ表示 : 既に定義されているグローバル変数やローカル変数などを表示する領域で、専用のウィンドウをマウスで作成することができる。この領域へのデータ表示は任意の入力項目と同期させることができる。

4.3 ボタン : 五つまでのボタンが設定でき、それぞれにファンクションキーと同様な機能を与えることができる。ボタンの形は、楕円形のものに限定されている。アプリケーションでのボタンの操作はカーソルキーの移動またはマウスのクリックで行なう。

4.4 ファンクションキー : 12個のファンクションキーを25行目に表示し機能を与えることができる。この部分の設計は会話型ではないが、サブルーチンにモデルが設定されているので、作成は容易である。

4.5 ベースウィンドウとタイトル : マウスを使用してサイズとバックグラウンドカラー、フレーム、タイトルの位置、文字列、文字色などを指定できる。

4.6 モジュールプロセジュアー : 各入力項目や表示項目などに対応する、データ処理はメインルーチンおよびサブルーチンの項目番号に対応する行番号以下にMで記述する。

5. 使用結果

画面設計は極めて容易で、カラーフルで機能的な画面が1時間程度で完成する。このシステムはルーチン全体の完全な自動作成ツールではない。しかし、メイン/サブルーチンのフレームは自動的に与えられるので、プロセジュアーを記述するのに、高度な技術を必要としない。M言語の初心者も苦勞なくプログラミングできる。また、画面作成の直後から、実行可能で、サブルーチン内のプロセジュアーを追加する度にそのままデバッグのためのテスト実行を行なえる。高度なG U Iが要求される今日、画面設計、マウスの操作、カーソルの移動、エディットリード、などにM言語を用いての記述を省いてプログラム作成できることは、システム開発の作業を画期的にスピードアップさせる。

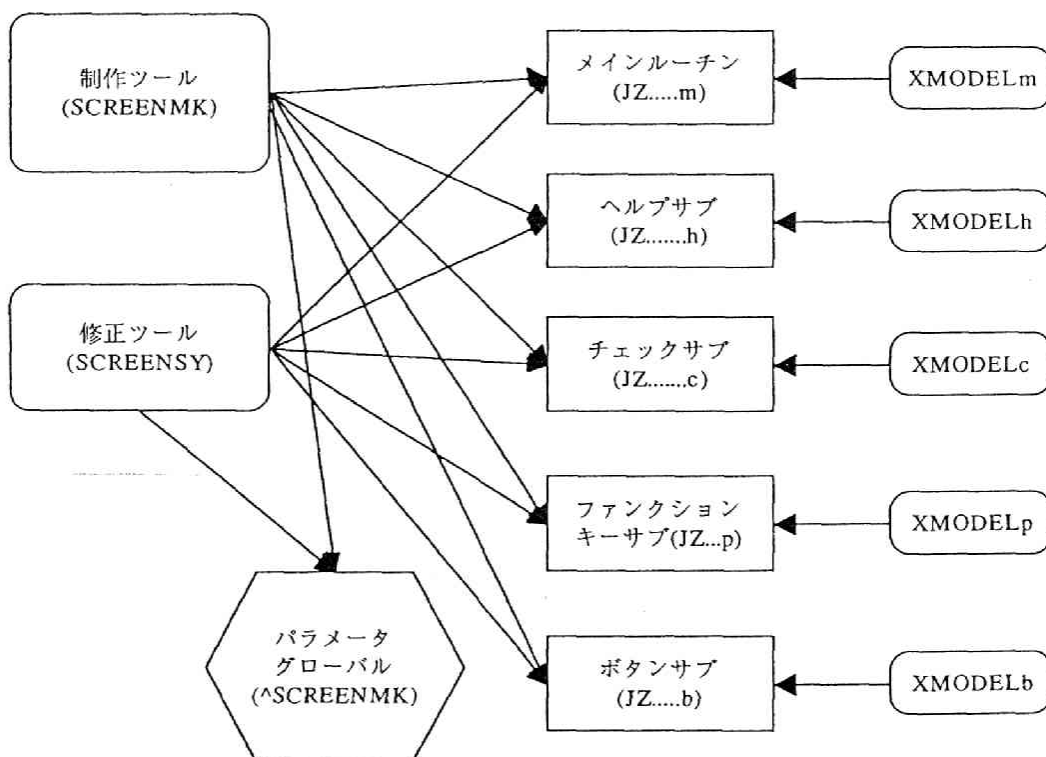


Fig.1 ルーチン作成過程

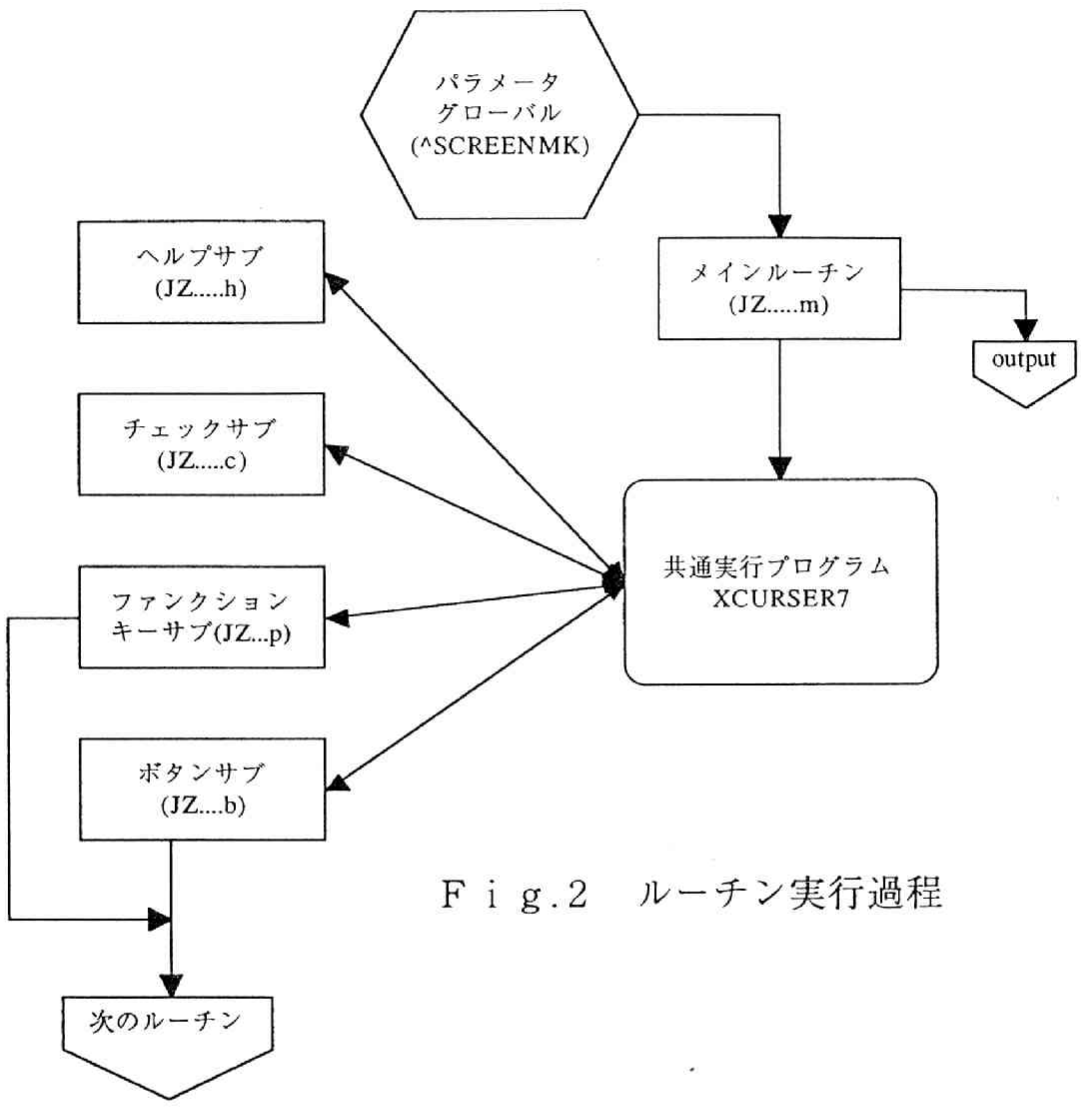


Fig.2 ルーチン実行過程

汎用シェルと医療向けシェルの知識表現と推論機構

Knowledge representation and inference engine
in general expert shell and medical expert shell

今泉幸雄、大櫛陽一
Yukio Imaizumi , Yoichi Ogushi

株式会社 SAS インステュート ジャパン
SAS Institute Japan Ltd.

- * 東海大学医学部病院管理学教室
- * Division of Hospital Administration ,
School of Medicine , Tokai University

知識表現として、汎用シェル向けにプロダクション・ルール（以下ルール）と医療向けにフレーム知識を用いた。これらの知識表現に対して、推論機構ではルールベースのエンジンを作成した。まず、全ての事実データに対して、前向き推論が働いた後に、後向き推論が動作して利用者への問い合わせをする。汎用シェルでは中間仮説を展開しながら推論する。医療向けのフレーム知識は推論される前までに、1スロット知識が1ルールに展開される。利用者からみると異なる知識表現に対して、同じ推論機構を採用した実例を報告する。

キーワード：知識表現、プロダクション・ルール、フレーム、推論機構
前向き推論、後向き推論、汎用シェル、医療向けシェル

In this expert shell system , there are production rule (referred as rule) for general expert shell and frame representation for medical shell. In this inference engine , we have made a rule base engine for each of rule and frame. Once the inference engine invokes the forward engine to all data , it invokes the backward engine to request to users the raw data. The general expert shell uses medium hypothesis during inference. Before the medical expert shell invokes the inference engine , it translates 1-slot knowledge to 1-rule. We hereby report , there is a different knowledge representation at viewpoint of users , but that these shell use the same inference engine .

Keyword: knowledge representation , production rule , frame , inference engine , forward engine , backward engine , general expert shell , medical expert shell

〒104 東京都中央区明石町6-4 ニチレイ明石町ビル 電話 03(5565)8391
Nichirei Akashicho Bldg. 6-4 Akashicho Chuo-ku Tokyo 〒104
Tel:03-5565-8381
〒259-11 神奈川県伊勢原市望星台 東海大学医学部 電話0463-93-1121
Bouseidai , Isehara-shi , Kanagawa-prefecture 〒259-11
Tel:0463-93-1121

1. はじめに

エキスパート・システムを作成するには、シェル（構築ツール）を用いるのが通常である。シェルは知識表現・推論機構・ユーザーインターフェース・知識ベース管理等から構成される。われわれは汎用シェル向けの知識表現としてルールと、ドメインシェル（特定専門向け）の対象として医療向けシェルのフレーム知識表現（以下フレーム）を作成した。推論機構としては異なる知識表現に対して、前向き推論と後向き推論を組合せしたエンジンを開発した。経緯としては先に汎用シェルを開発したので、汎用シェルのルール知識表現、推論機構、医療向けシェルのフレーム知識表現の順で以下に報告する。

2. 汎用シェルのルール知識表現

利用者がルールを編集（入力・変更・削除を総称して編集と呼ぶ）する時は、知識ベース編集を実行する。利用者がルールを管理しやすくする目的で、グループ知識（内部ではメタ("META")）を用いる。1グループ知識は複数のルール集合より成り立っている。グループ知識単位に"診断型"と"計画型"の区別をする。

2.1 知識ベース編集

編集の単位は1グループ単位で、かつグループに属する1ルールが対象となる。1ルールは"もしも***ならば、***の疑いがある"というIF（条件部）、THEN（結論部）の表現を採用し、条件部は複数の表現が可能である。結論部で結論名に対する確信度を入力する。1ルールの表現としては3種類存在する。

- (1) 条件部が事実データで結論部が中間仮設名
- (2) 条件部が事実データで結論部が結論名
- (3) 条件部が中間仮設名で結論部が結論名

下記に実際の実行例を示す。

1. 知識ベース編集(ルール知識の編集)

ルール知識名は"グループ知識名(英数字5文字以内)+"数字(3文字以内)"からなる。大きくは条件部(if:もしも***ならば)と結論部(then:***の疑いがある;確信度は0.7)から構成される。条件部の日本語は必ず主語+格助詞(はorが)+動詞(例:鼻水がある)とし、結論部は名詞句(風邪の疑い)に確信度は $0.0 < x < 1.0$ あるいは"YES"となる。

1. ルール知識名(グループ知識名+数字) : YIKB010
2. 結論部の名詞句(英数字20文字以内) : 風邪の疑い
3. 結論部の確信度 : 0.7
4. 条件部1(英数字30文字以内) : 鼻水がある
5. 条件部2(英数字30文字以内) : 熱が38度以上である

2.2 知識ベースファイル

知識ベースファイル(^ZFKB)はグループ知識("META")とルールに分けている。ルール名は"グループ知識名+数字"から構成されているので相互に参照できるようになっている。推論実行の初期処理としてZFKBは、1ルールの条件部の一部をキーとした知識テーブル(ZTKB)として展開される。

```

^ZFKB
| ★ 設定情報
| ->>^ZFKB(KBFLNM) :
|   |   ①KEY: 知識ベースファイル名  VX(8) (標準名として"DEFAULT")
|   |   ★ : 知識ベースファイル情報
|   | ->>^ZFKB(KBFLNM,"META") :
|   |   |   ②KEY: Meta知識の定義  FA(4)
|   |   |   |
|   |   | ->>^ZFKB(KBFLNM,"META",METANM)
|   |   |   |   ③KEY: 個々のMeta知識の名前  VX(6)
|   |   |   |   |
|   |   |   | ->>^ZFKB(KBFLNM,"META",METANM,"NMDATE")
|   |   |   |   |   ④KEY: 作成定義1  FA(6)
|   |   |   |   |   ★ : 作成定義1情報
|   |   |   | ->>^ZFKB(KBFLNM,"META",METANM,"COMENT")
|   |   |   |   |   ④KEY: 作成定義2  FA(6)
|   |   |   |   |   ★ : 作成定義2情報
|   |   |   | ->>^ZFKB(KBFLNM,"META",METANM,"KBBODY")
|   |   |   |   |   ④KEY: 作成定義3  FA(6)
|   |   |   |   |   ★ : 作成定義3情報
|   |   |   |
|   |   | ->>^ZFKB(KBFLNM,"RULE") :
|   |   |   |   ②KEY: ルール知識の定義  FA(4)
|   |   |   |   |
|   |   |   | ->>^ZFKB(KBFLNM,"RULE",RULENM)
|   |   |   |   |   ③KEY: 個々のルール知識の名前  VX(8) (METANM+
|   |   |   |   |   ★ : ルール知識の内容          +"1" ---> "n" )

```

3. 推論機構

メニューから推論実行が選択されると、推論の初期化として知識テーブル(ZTK B)を作成して、利用者から事実データの入力をうながし、事実データの全てに対して前向き推論が実行される。前向き推論結果が表示された後に、後向き推論が実行されて、利用者に対してまだ入力されていない事実データの値の入力を要求して推論する。後向き推論が終了すると両推論の結果が表示される。下記に推論の実行例を示す。

3. 推論実行

今から推論をする準備をします。

知識ベースのファイル名を入力してください : KB001

ルール数は20で、作成者は鈴木一郎で、

作成日は1991年5月4日で、最新更新日は1991年5月10日です。

事実データは主語、助詞("は","が"),述語の日本語とする、

例として[熱は高い , 鼻水が多い]の形式となる。

ADD/DATA(事実データの追加:終了は改行キーのみ押す) : 下病は米水

ADD/DATA(事実データの追加:終了は改行キーのみ押す) : 熱は39度以上

ADD/DATA(事実データの追加:終了は改行キーのみ押す) : 下病の頻度が多い

ADD/DATA(事実データの追加:終了は改行キーのみ押す) :

つぎの処理を選択してください

ADD/DATA,SHOW/DATA,RUN(推論) : SHOW/DATA

事実データの内容です。

1. 下痢は米水
2. 熱は39度以上
3. 下痢の頻度は多い

以上3件である。

ADD/DATA,SHOW/DATA,RUN(推論) : RUN

これから(前向き)推論を開始します。

中間仮設として ==> 風邪の疑いがある。

用意された事実データからの推論が終り、

これから(後向き)推論を開始します。

涙目があるの症状はありますか (Y/N/<U>) : W	Why (理由)
花粉症の確認をします。	
涙目があるの症状はありますか (Y/N/<U>) : T	Trace on
(trace: 質問生成の処理を実行してます)	
涙目があるの症状はありますか (Y/N/<U>) : E	Explain (説明機能)
ルール番号(RB101)が起動されてます。	
涙目があるの症状はありますか (Y/N/<U>) : N	No
(trace: 質問生成の処理を実行してます)	
咳はでるの症状はありますか (Y/N/<U>) : W	Why (理由)
風邪の確認をします。	
咳はでるの症状はありますか (Y/N/<U>) : T	Trace off
咳はでるの症状はありますか (Y/N/<U>) : Y	Yes
脱水があるの症状はありますか (Y/N/<U>) :	Unknown

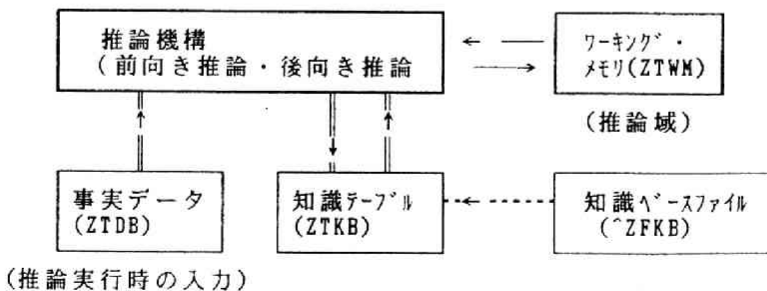
推論結果は以下のとおりです。

1. 風邪である。 1.00
2. コレラの疑いがある。 0.75

推論の実行は終了しました。

If any key return , we will come to previous menu.)

推論のメカニズムの概略としては次の図になる。



推論域となるワーキング・メモリー(ZTWM)を中心に説明する。ワーキング・メモリーに格納された候補数は推論終了したときに表示される候補数となる。推論中に新しい候補名が入ると"+1"(1加算)される、初期値は"0"である。知識ルールが推論実施されると、ワーキング・メモリー(ZTWM)に出力される、例えば下記の知識テーブル(ZTKB)が事実データテーブル(ZTDB)あるいはZTWMと一致すると

```
ZTKB("RB204")="NOTRUN"^^"YES";"下痢"$が"$"米水"$"YES"^^
      コレラ"^0.7"^^"RRA30";
ZTWM("コレラ")=2;RF305,YES^RB204,0.7
```

この場合は"コレラ"はすでに候補として知識ルールの"RF305"も一致していた。全ての推論の実施終了時にワーキング・メモリー(ZTWM)の内容が表示(一部確信度などは計算する)されて推論の終了となる。

ZTWM

| ★ 設定情報

|

| ->>ZTWM(INFCNDNM):

①KEY: 知識ルールを推論した結果の候補名(例として"風邪の
疑い","コレラ") VX(10)

★ : 推論した候補名に対する参照頻度や知識ルール名などの情報
(★: データノードを示す。)

4. 医療向けシェルのフレーム知識表現

医学教科書は疾患名(病名)に対して考えられる複数の症状(データ)を並列的に記述されているものが多い。知識の編集単位も疾患名単位のが扱いやすいのでフレームで表現する。知識の入力方法としては疾患名(結果 OR 結論)を先に入力してから、その症状名(原因 OR 条件)を複数個入力(n個)する。

4.1 フレーム知識表現の解釈

(例) コレラや風邪は疾患名と考える。

コレラ ----- 発生頻度 1.0% (事前確率)

+

+-----> 熱が39度以上 10% (事後確率)

+-----> ふんが米水の状態 60%

+-----> 皮膚に赤い湿疹 10%

+.....

(注意)事前確率<==公衆衛生学と社会医学: 季節などに変化する。

事後確率<==臨床医学: 医学上の定説。

疾患名と症状の因果関係は基礎医学(病理など含む)。

疾患名単位で入力された知識は、実際に推論をするまでには、ルール単位に変換されていることである。上記の疾患名の"コレラ"フレームは3つのスロット知識からなっていると解釈し、3つのルールに変換される。確信度は事後確率*0.01*事前確率として計算する。上記の例としては熱が39度以上あればコレラの疑いは0.6(=60*0.01*1.0)という1ルールになる。1ルールは条件部、結論部は一つになる。

4.2 フレーム単位の知識編集

疾患名単位をグループ知識（あるいは"META"）として次のように入力をする。

1. 知識ベース編集(グループ知識)

1. グループ知識の名前 : YIKB
2. 結果(結論)(英数字10文字以内) : cholera
3. 発生頻度(事前確率) : 1.0
4. コメント(日本語30文字以内) : 冷凍の魚介類の輸入品
5. 参考文献(日本語30文字以内) : 医学概論 (at page 120)
- ***条件(日本語20文字以内: "主語"+"は"or"が"+"述語")と事後確率(1から100の間)***
6. 条件: 熱が39度以上 確率: 10
7. 条件: 下痢がひどい 確率: 45
8. 条件: 皮膚が赤い湿疹 確率: 30
9. 条件: ふんが米水の状態である 確率: 55
10. 条件: 確率:
11. 条件: 確率:

4.3 知識ベースファイル(医療向けシェル)

フレームを格納する時に、グループ知識名の下位構造にグループ共通情報となる部分("META")とルール単位("RULE")に分けられる。ルール名は知識編集にて保存を指示すると自動的に"グループ知識名+数字"という名前で作成される。下記に知識ベースファイル(^ZFKB)の構造を示す。

^ZFKB

| ★ 設定情報

| ->>^ZFKB(KBFLNM) :

| ①KEY: 知識ベースファイル名 VX(8) (標準名として"DEFAULT")
| ★ : 知識ベースファイル情報

| ->>^ZFKB(KBFLNM, METANM) :

| ②KEY: 個々のMeta知識の名前 VX(6)
| グループ知識名を^ZFKBでは、メタ知識名(METANM)と

| ->>^ZFKB(KBFLNM, METANM, "META", "NMDATE")

| ③KEY: 作成定義1 FA(6)

| ★ : 作成定義1情報

| ->>^ZFKB(KBFLNM, METANM, "META", "COMENT")

| ③KEY: 作成定義2 FA(6)

| ★ : 作成定義2情報

| ->>^ZFKB(KBFLNM, METANM, "META", "KBBODY")

| ③KEY: 作成定義3 FA(6)

| ★ : 作成定義3情報

| ->>^ZFKB(KBFLNM, METANM, "RULE", RULENM)

| ③KEY: 個々のルール知識の名前 VX(8) (METANM+
| ★ : ルール知識の内容 +"1" ---> "n")

5. おわりに

今回は異なる(フレームとルール)知識表現にたいして、同じ推論機構が動作したことを報告した。エキスパートシステムの利用者からみると、自分の専門分野での知識・情報がコンピュータの表現と自分の頭の中の表現との差が少ないのがよいことである。本報告にて、利用者が汎用の知識表現やドメイン知識表現を選択して利用すればよい事が提案できたと思う。汎用シェルは医学部学生の91・92年度"知識工学"の授業の演習として採用し、医療向けシェルは同じく93年度の授業演習として採用予定でいる。次の課題としては医療向けシェルの評価をすることと知識表現の中の日本語表現の類似性の認識をどう扱うかである。

《参考文献》

- (1) Richard Fikes and Tom Kehler : "The rule and Frame-based Representation in reasoning" , Communications of the ACM , September 1985 , Volume 28 , Number 9 , pp904-920(1985)
- (2) M.Minsky : "A Framework for Representing Knowledge , in the Psychology of Computer Vision" (McGrawHill) (1975)
- (3) Koyama T. : "An Automatic Rule Generation and Evaluation Schema for AI Medical consultation systems" , 3rd Medical Informatics , pp1306-1310(1980)
- (4) 今泉幸雄 他 : " 実験動物データ解析の支援システム(1)" , 情報処理学会 , 第31回(昭和60年後期)全国大会論文集 , pp929-930(1985)
- (5) 今泉幸雄 他 : " MUMPSによるデータ解析のエキスパートシステム" , PCworld , 1988 11月号 , pp68-74(1988)
- (6) 今泉幸雄 : " 知識の矛盾抽出と修正法 -時制論理の考察-" , 第15回日本MUMPS学会(名古屋) , pp153-159(1988)
- (7) 今泉・大節 : " エキスパート・システムにおける知識表現と推論機構" , 第18回日本MUMPS学会(伊勢原) , pp133(1991)

自然語情報をデータベースに取込むための構造言語学の応用
Application of Structured Linguistics to Organize Natural Language into the DBs

マルシオ ビチキ ド アマラル、里村 洋一

Marcio Biczyc do Amaral, Yoichi Satomura

千葉大学医学部附属病院医療情報部

Div. of Medical Informatics, Chiba University Hospital

The most used way for input of information in the DBs is through the use of pre-formed screens that the user must fulfil. For organization of natural language information other techniques are required. We have been working with semantic parsers for analysis of texts. In this research we explore a new path for data formatation, that is based on the application of structured linguistic for the organization of patient case reports. We use six semantic types that are: "D" for Diseases/Diagnoses, "H" for History, "S" for Symptoms, "P" for Physical Findings, "T" for Tests and Laboratorial Results, and "R" for Treatment. Associated with these semantic categories we use six types of symbols: comma ",", colon ":", semi-colon ";", asterisk "*", parenthesis "()", and point ".". The categories and symbols are combined in order to organize the texts in a format similar to the one that will be set into the database. The advantage of this method is that the parser will effectively work for virtually all cases since the text is already structured in a database format.

1. Introduction

The use of natural language in most Electronic Medical Records (EMR) and other areas is restricted because of the complexity related to the field. The solution used by far is to restrict the input of information using menus or rigidly formatted programs' screens. Some institutions permit free entrance of natural language descriptions of patient cases as described by the doctors, and indexing of the text using a few keywords. The text can be retrieved using only the keywords, or information like the patients' number, date, etc. In the other extreme of the techniques is the natural language parser. The purpose is to extensively analyse the natural language statements. There are syntactic based, semantic based, mixed and other types of parsers. The syntactic approach is older, but it is still very used [1,2]. The application of semantic approaches is more recent [3,4,5]. Both methodologies however do not account for processing different kinds of texts, with high accuracy. Indeed, actually there are no programs that can really understand natural language. However, if you restrict the applications domain, good results can be obtained. We investigate the possibility of implementing one program for input of patient case reports, that does not use pre-formed screens, nor a natural language parser. We describe one solution, that can be placed in the middle of these two extremes. We apply structured linguistic to the medical field, permitting the structuration of the medical text into a database format.

2. Methodology

We developed a notation to implement an structured grammar. This notation is composed of six medical semantic categories, and six symbols. The semantic categories are: D for Disease/Diagnosis; S for Symptom; P for Physical Examination; H for History Data; T for tests or lab data; R for Treatment. The symbols are: 1) comma ","; 2) colon ":"; 3) semi-colon ";"; 4) parenthesis "()"; 5) asterisk "*"; 6) point ".". These symbols will be used combined with the semantic categories to describe unambiguously the medical text. The user have to learn the rules for interpretation of the medical text. There are three basic rules: First: identify the medical term(s) in the statement; Second: identify the attributes related to these terms; Third: identify the type of relation between the medical terms. Using this simple rules the user will enter first the code of the semantic category, p.e. "S", that implies the description of symptoms. Between parenthesis "()", the user will enter the related attributes. To represent a relation "*" is used. We describe one example of how a natural language text is transformed into a database using this method. Figure 1 shows one example of a patient case report extracted from a journal.

Case Record from the New England Journal of Medicine, Case 42-1992

Natural Language Text: A 59-year-old woman was admitted to the hospital because of progressive dyspnea. The patient was in excellent health until five years earlier, when she began to experience exertional dyspnea that gradually worsened. Seven months before admission she discontinued a 50-pack-year habit of cigarette smoking, with transitory improvement. Three months later the dyspnea began to worsen progressively and was accompanied by fatigue, and she entered another hospital. An x-ray film of the chest showed a diffuse, bilateral interstitial infiltrate. A ventilation-perfusion (VP) scan indicated a low probability of pulmonary embolism. An USG study of the heart, with Doppler technique, disclosed that the size and function of the left ventricle were normal. A transbronchial biopsy yielded a specimen that was normal on microscopical examination. A limited pulmonary angiographic examination demonstrated no evidence of pulmonary emboli.

FIGURE 1: The medical doctor describes the patient case using natural language. The medical scientific sublanguage is embeded in natural language statements. The analysis and comprehension of this text by a NL parser requires complex procedures.

In order to simplify the work natural language processing, the user will read this text, select the medical words, attach an semantic label identifying the medical term as a symptom, a diagnosis, etc, identify the attributes and relations and described the interpreted statements using our notation. Usually, the medical terms that have to be identified are nouns. The attributes are adverbs, adjectives or other modifiers. Typical modifiers for a symptoms, for example, are time (1 day, 2 months, 5 years), type (progressive, cyclic), grade (important, moderate, mild). See figure 2.

<i>Input using our method:</i>	<i>Text Structure</i>
S: <u>dyspnea</u> (progressive).	S: _0.
H: <u>health</u> (excellent)(5 years earlier);*S: <u>dyspnea</u> (exertional) (worsed).	H: __00;*S: __00
H: <u>smoking</u> (stop 7 months before)(50-pack-year).	H: __00.
S: <u>dyspnea</u> (3 months later) (worsen progressively).	S: __00.
S: <u>fatigue</u> ; *H: <u>hospital</u> (another) (entered).	S: _; *H: __00.
T: <u>x-ray</u> (chest)(diffuse bilateral interstitial infiltrate).	T: __00.
T: <u>VP scan</u> (pulmonary embolism) (low probability).	T: __00.
T: <u>Doppler-USG</u> (heart)(left ventricle)(size and function normal).	T: __000).
T: <u>Biopsy</u> (transbronquial) (normal).	T: __00.
T: <u>Angiography</u> (pulmonary) (no evidence of emboli).	T: __00.

FIGURE 2: This figure shows the format of input of the natural language text described in figure 1. The symbol "S" means symptom, "H" means history, "T" means test. After describing the semantic category followed by a ";", the user describes the identified medical term, for example "dyspnea". Between parenthesis "()", the user enter the attributes related to the medical term, for example, "progressive", "5 years earlier". On the right side of the figure, the abstract structure of the text is represented. The program will have only to analyse this structure to parse the text.

The next step is the structuration of the text using a M global. The database format will be organized using the patient number, the clinic, the date and time as administrative data. The semantic categories will be used to identify the class of the information, followed by the specific medical term. The attributes will be the values of the global. See figure 3.

Format and Contents of the Structured Database:

```

^NLEMR(patient number,clinic,date,time,S,dyspnea)="progressive"
^NLEMR(patient number,clinic,date,time,H,health)="excellent^5 years earlier"
^NLEMR(patient number,clinic,date,time,H,health,S,dyspnea)="exertional^worsed"
^NLEMR(patient number,clinic,date,time,H,smoking)="stop 7 m before^50-pack-year"
^NLEMR(patient number,clinic,date,time,S,dyspnea)="3 m later^worsen progressively"
^NLEMR(patient number,clinic,date,time,S,fatigue)=" "
^NLEMR(patient number,clinic,date,time,S,fatigue,H,hospital)="another^entered"
^NLEMR(patient number,clinic,date,time,T,x-ray)="chest^diffuse bilat interstitial infiltrate"
^NLEMR(patient number,clinic,date,time,T,VP-scan)="pulmonary embolism^low probability"
^NLEMR(patient number,clinic,date,time,T,Doppler-USG)="heart^left ventricle^size,func normal"
^NLEMR(patient number,clinic,date,time,T,Biopsy)="transbronquial^normal"
^NLEMR(patient number,clinic,date,time,T,Angiography)="pulmonary^no evidence of emboli"

```

FIGURE3: This figure shows the structured information as organized in a M global.

3. Discussion

We developed this conceptual method in order to define a technique that could be an intermediate between the pre-formed screens and the pure natural language parsers. To interpret the entered structured information, the program can perform a nonambiguous analysis of the information, since all grammatical rules are known by the program. One problem is related to the need for the user to learn how to identify the medical concepts and enter the information using the adequate format. This is not a difficult step, since we use only six types of semantic categories and a set of only six symbols. This solution can be useful if we want a method that can be used to organize natural language medical text. We have been working with natural language parser, and the function of the parser is exactly transform the text in the format we described in figure 3. This procedure can be performed if the analysed domain is restricted. Using the methodology described above it is almost always possible to build the structured database, because the interpretation of the text and structuration process is performed by the human user. The method can be used for any natural language, like Japanese, English, Greek, since the grammatical rules used by the parser will not depend on any specific language. This process is different regarding parsers, that will have to use specific syntactical rules depending on the analysed language. Another advantage is the versatility of the method that can be adapted to different medical case reports. It is not dependent on the sequence of the entered facts. For example, it is not necessary to enter first the history data, followed by the symptoms, the physical findings and the lab data or tests, the concluded diagnosis and the treatment. In real case reports, as exemplified in figure 1, the text can describe the medical facts following the time dimension, alternating symptoms, history data, treatments, etc. This method permits the input of text independent of the sequence of the medical events. The temporal information is stored together with the medical term which it is associated, permitting the storage of the information in a temporal sequence.

REFERENCES

- [1] King M (ed). Parsing Natural Language. London: Academic Press, 1983.
- [2] Pereira F, Warren D. Definite clause grammars for natural language analysis. In Artificial Intelligence 1980, 13:231-278.
- [3] Winograd T. Understanding Natural Language. 1972. Edinburg University Press.
- [4] Do Amaral MB, Satomura Y. Semantic Patterns to Represent Medical Information into a Language Independent Database. In Proc of the 7th Japanese Congress in Artificial Intelligence, 1993.
- [5] Baud RH et al. Natural Language Processing and Medical records. In Medinfo 92, pp 1362-1367.

M言語 と Medical Decision Making

M Language and Medical Decision Making

鎌江伊三夫、笹川紀夫、林振健、山本和子

島根医科大学医学部医療情報学教室

Isao Kamae, Norio Sasagawa, Lin Zhen Jian, Kazuko Yamamoto

Department of Medical Informatics, Shimane Medical University

はじめに

1970年代の前半以降、米国を中心として医学意志決定(MDM: Medical Decision Making)と呼ばれる医学・医療における科学的意志決定法の研究が進展し、医療情報学的な観点からもその重要性が増加しつつある。本論では、M言語がこのMDMの基本手法のプログラミングや教育において有用であることを述べる。

MDMの基本手法

医学的な意志決定を行なうための選択枝がいくつかある場合、「それぞれの選択枝を選んだ場合の期待効用を比較計算して、その期待効用が最大となる選択枝を選ぶ。」というのがMDMの基本手法である。例えば、図1は、腸間膜動脈塞栓症か急性腸炎かを疑う急性腹症の場合、開腹手術を行なうべきか否かのMDMの決定樹を示している。長期生存率を効用として採用し、開腹手術を行なう場合とそうでない場合の期待効用を計算すると、それぞれ0.8と0.765と

なる。従って、手術を行なわない方がよいとの結論を得る。

決定樹のM言語表現

図1が示すように、MDMにおいては決定樹の生成が期待値計算に必要であり、この基本手法とそれをプログラミングする観点からは、樹構造が柔軟に扱えることが重要である。MDMを実行するソフトウェアは、既にいくつか開発されているが、M言語における樹構造データベース構築の容易さは、M言語をMDMに対して特に良好な親和性をもつものとしている。例えば、「手術を受ければ85%の確率で生存し、そのとき急性腸炎である確率は80%であり、その場合の長期生存率は1.0である。」という決定樹に沿う知識は、適当なグローバル変数と添字を用いて、

```
OPERATION("do", 85, 80) = 1.0
```

と簡単に表現できる。従って、M言語によるプログラム:

```
OPERATION("don't", 20) = 0
```

```
OPERATION("don't", 80) = 1.0
```

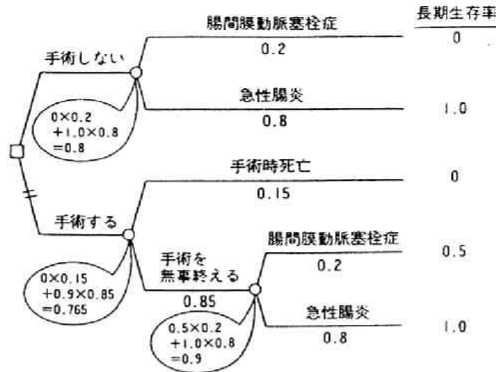


図1. 急性腹症の決定樹の例

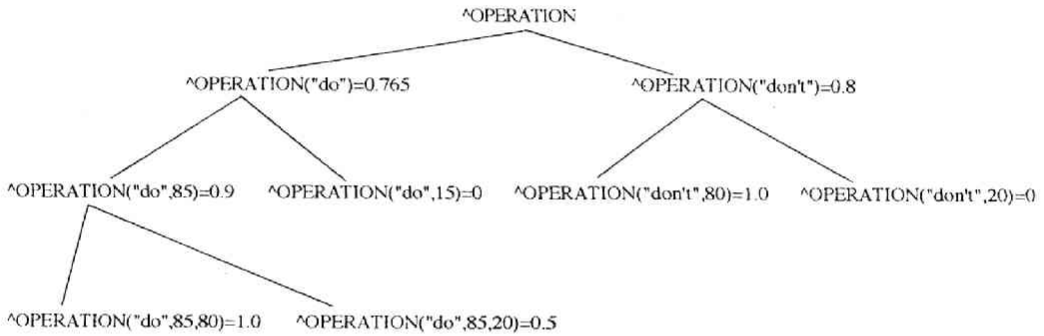


図2. 図1に等価なM言語による樹構造

$\hat{OPERATION}("do", 15) = 0$
 $\hat{OPERATION}("do", 85, 20) = 0.5$
 $\hat{OPERATION}("do", 85, 80) = 1.0$

は、図2に示される樹構造データベースを生成する。これに対し、若干の系統的演算を行なえば、容易に図1のMDMの決定樹と等価な論理構造を実現できる。

能性を示唆する。一般医師、医学生などへの初級プログラミング教育においても、MDMの基本手法の修得も同時に期待する場合に、特に有用であると考えられる。

文献

まとめ

MDMにおける決定樹のプログラミングが、M言語を用いて容易に実現できることを示した。このMDMとM言語の関係は、発展的なMDM用のソフトウェア開発の可

1. 桜井恒太郎：医学判断学. Therapeutic Research 9:92-102, 1988.
2. 久繁哲徳編：臨床判断学. 篠原出版 東京、1988.
3. 福井次矢：臨床医の決断と心理. 医学書院 東京、1988.

個人病院に於ける病院情報システムの活用

— レセプト専用機からMUMPSマシンに変更して —

○土屋 喬義（獨協医科大学・第一小児科、土屋小児病院）
土屋 恭子（土屋小児病院）
木村 一元（獨協医科大学・総研ME）

医療法人土屋小児病院は、埼玉県北東部の久喜市に所在する1日平均外来患者数230人、病床数22床の外来診療を中心とする小病院である。標榜科目は小児科を中心とし、同時に皮膚科、内科が併設されている。

今日、中小病院の経営悪化が取りざたされているが、小児科領域では、さらに近年の少産傾向による小児人口の減少、患者の専門指向、大病院指向により中小の外来診療を中心とする医院、病院（現在でも小児科を主体としている病院は非常に少ない）の経営は大変苦しいものがある。このため、この厳しい環境に打ち勝つべくM（マンプス）言語を使用した病院情報システムパッケージを導入し患者サービスの向上を図った。病院情報システム導入後患者の待ち時間の減少は著しく病院の患者数の減少傾向は無くなり、患者数は増加に転じている。今後、検査システム、病棟のオーダー入力システムを採用し業務の合理化を行っていく予定である。

はじめに：

当院は、土屋小児科医院として約25年前に開設して以来、ベビーブームの波に乗り順調に患者数を増やしていき、13年ほど前より、現在の医療法人土屋小児病院となった。しかし、上述したような環境の変化によりここ5年半ほど前より患者数の減少傾向を認めている。患者数の減少傾向は小児人口の減少の為のみならず、近隣の新しい小児科を標榜する医院の開業による相対的なサービスの悪化、当院の医師の高齢化による診療能力の低下などが考えられた。この状態を解決するためには、特に待ち時間が重要と考え、これに対してコンピュータを活用し改善しようと試みた。

病院情報システム採用前：

当院では、15年ほど前よりレセプトコンピュータを用いていた。1代目のマシンは大型機のデータ入力端末で入力したデータをカセットテープに出力し、銀行の大型機で処理し、レセプトを清書するもの、2代目（12年前）、3代目（8年前）は沖電気のレセプタシリーズを使用した。3代目のレセプトコンピュータはオフコンベースのもので、端末が2台使用できた。入力はキーボードの無いコードセレクトを使用するタイプで、初心者にも使用しやすい利点はあったが、半面、ある程度慣れると入力速度が思ったほど早くない、端末の反応が鈍く入力に時間がかかる、コボルで書かれておりユーザー側にデータ及び開発環境がオープンにされておらず業務内容に合わせて思うようにカスタマイズできない、検索速度が遅く患者の氏名検索を行うともう1台の端末の速度が更に低下するなどの欠点があり、そしてなによりマン・マシンインターフェースが悪くベテランでも入力速度が上がらなかった。

この様な状態では

1) 来院時もし患者が受診券を忘れた場合、カルテを番号で管理しているのでカルテ番号を探すためには、遅いコンピュータの検索プログラムを起動する（同時に動いているレセプト入力ジョブをほぼ停止に近い状態にしてしまう）。または、あらかじめ作ってある索引ノートより検索する。この作業のために受付業務が停滞し、たたさえ長い診察までの待ち時間がさらに長くなり、また、この検索のために他の部所の者が借りだされ受付が混乱した。

2) 患者1名当たりのレセプトデータ入力時間が2分以上かったため、（手計算では

1分以内) 1日の外来患者をすべて診療時間内に入力する事は難しく、どうしても忙しい時は手計算となり後で残業としてレセプトデータを入力する事となっていた。このため同じ事を手計算とコンピュータによる計算の2回行う事となりコンピュータは単なるレセプト清書マシンとなっていた。また、この2重の仕事のため事務系の職員の余裕が無くなり、会計計算が遅れ診療後の待ち時間も更に長くなっていた。

3) 処方箋の発行は医師が直接書くか、または介助者に口述筆記してもらい印鑑を押さなければならない。処方内容の記載は処方箋とカルテの両方に記載をすることとなり2度手間となっていた。このため看護助手に処方箋を写してもらったり、複写の処方箋を用意しているのが実状で、この労力も大変なものであった。

4) 薬袋に名前と用法を書き入れる仕事は、患者さんの集中する午前中は手伝いの看護学生が完全にかかりきりになる仕事であった。

改善の計画：

以上の様な状況を打破し、事務系の仕事をスムーズにするため医療情報システムの導入を計画した。

システム導入に際しては、オフコンを含む様々なシステムを検討したが

- 1) 病院の事情に合った環境が構築できること。
- 2) 少なくとも5年以上陳腐化しないこと。
- 3) ユーザー側でもファイルの共有、ソフトウェアの作成ができること。
- 4) 基本的なソフトウェアが優れていること。
 - a) ユーザーインターフェースに優れ、十分な入力速度が得られること。
 - b) 導入に際し入力負荷以上に仕事を合理化し、可能ならば事務部門の人員を減らせる事ができること。
 - b) 病院の事情に合せカスタマイズできること。
- 5) ベンダーのサポートが優れていること。

以上の条件を考え、M言語により構築された住友電工システムエンジニアリングのACCELを採用した。

改善の方法：

カルテの流れを中心に徹底的に時間のかかる要素を洗い出し、コンピュータで出来る仕事を抽出した。その結果、以下の改善が必要と思われた。

- 1) 再診の患者のカルテの抽出速度を上げること。

小児科では長期に受診を続ける患者が少なく、また、年長となると小児科を卒業し、内科に転院し、当院を離れる患者も多い。そこで、1年間受診しない患者のカルテは、インアクティブカルテとし2次保管とした。アクティブカルテの管理方法は、今までのランダムな番号方式よりターミナルデジット方式に改めた。これを名前、生年月日、患者番号より検索させるためのソフトのカスタマイズをベンダーに依頼する事とした。

- 2) 診療の終わったカルテは直ちにコンピュータに入力されること。

当院は通常3人の医師が同時に診療に当たっているが、一人の患者の診療時間が忙しいと3分以内になってしまうことがある。このため事務の職員は、診療が終わったカルテを1分以内に入力しなければ、次々にカルテが溜ってしまい3分間診療、会計1時間待ちとなり大変評判が悪くなる。このためには如何に早くレセプト情報をコンピュータに入力できるかが問題となる。入力速度を上げるためにはキーボードに慣れることは重要なことであるが更にメニュー等の構成ファイルを適性化し如何にキーストロークを減少させる様にカスタマイズできる事も重要と考えた。

- 3) 入力した情報で再利用できるものは最大限利用する。

入力したレセプト情報は処方箋、薬袋の名前に反映させる事により作業の効率化を図った。当院での診察室で発生するオーダーと事務室で処理するレセプト入力との差を調べると初診、再診の区別以外同一の処理であることが判明した。そこで事務室にある端末を各

診察機の隣に置き、事務職員（診察室に出入りするようになるのでメディカルクラークと呼ぶ事にした）がオーダーの入力を行い同時に処方箋の代行入力を行う事とした。これにより高価なオーダー入力ソフトを購入せずに、また新しい入力補助者を雇わずに処方オーダーシステムを実現する事が可能となった。

また、同時に薬剤部門に置いたプリンターより薬袋の名前、用法、日数をラベルに打出す事とした。

結果：

今回当院にて採用した住友電工システムエンジニアリングのUNIXワークステーション上のM言語、U-MUMPSを使用した病院情報システムACCCELは大学病院、総合病院に採用されている大変大きなシステムである。このため私共の様な小規模で、システム管理者の置く事ができない病院でうまく運用できるかどうか、かなりの不安があった。また、これまで使用していたオフコンのタッチパネルからキーボードへとユーザーインターフェースが大幅に異なったため職員の訓練に多くの時間を費やさなければならなかった。新たなシステムへの移行のために2ヶ月間、新旧システムの平行稼働を行ったためこの間の事務処理能力はさらに低下し、かえって待ち時間の増加を招き、その後約3ヶ月間の患者数の減少を引き起こした。しかし、その後は操作の習熟による入力速度の上昇、省力化による処理能力の増大、そして作業の合理化により全体の待ち時間の減少を認めるようになった。特に診察終了から会計終了までの時間短縮が著しく約1時間待ちの所が平均20分までに短縮することが可能となった。また、診察の待ち時間も今までの50%程度に減少させることができた。さらに1日中薬袋の名前書きをしていた看護学生を本来のセクションに戻す事ができるようになった。

カルテ管理のターミナルデジット化とそれに合わせてカスタマイズした検索ソフトのおかげで再診患者の受け付け処理時間は、ほぼ予想通り減少し受け付けの職員の負荷は減少しかなり余裕がでるようになった。しかし初診者の受け付け時の入力項目が多く、以前手書きで行っていた受け付け処理より時間がかかり、この部分にかなりの時間を取られてしまっていることも現状である。

考案：

現在の医事システムは手書きのレセプトを基に発達してきたものであり、文字情報が多く、また、例外的な処理も非常に多くなっている。このため入力される情報も多く入力時間もかかり、人員の合理化も難しいと言わざるを得ない。このためOCRシステム等を導入しさらに入力負荷を減少させる必要性を感じた。

医師によるオーダーリングシステムの入力は、私共の病院の様に医師が忙しい外来の中で使用するのには大変難しい事だと思われた。第一に入力に時間がかかりすぎる。通常実質2-3分間の診療の間に診察をして、説明を行ない、カルテを記載し、処方箋を発行し、更に検査のオーダーや注射のオーダーも行なわなければならない。この間に医師が端末よりオーダーを入力する事になる。実際、試験的に私の外来でも処方コンピュータ入力及び病名の入力を試みたが、次の点に問題があると思われた。

1) CRTが巨大で診察機の大きな面積を占領する、カルテに記載する量が多いにも拘らず更に大量のキーストロークが発生する。

2) 通常、視点はカルテと患者の間の移動のみで済むところが更にCRTまで視点を移動しなければならないため、目が回って気分が悪くなる。

私共の様な病院でオーダーリングシステムを採用するためには、医師の机の上に視線を移動させずに済む位置に（出来れば机と同一平面上に）液晶等の表示画面上でペン入力が可能になれば少しは入力する気になれるかも知れない。また、看護婦が入力するためには発生源で入力するために簡単に持ち運べるハンディ端末が必要と思われる。現在、これらはまだ十分な性能の製品は開発されていないが、近いうちに開発されるものと期待している。

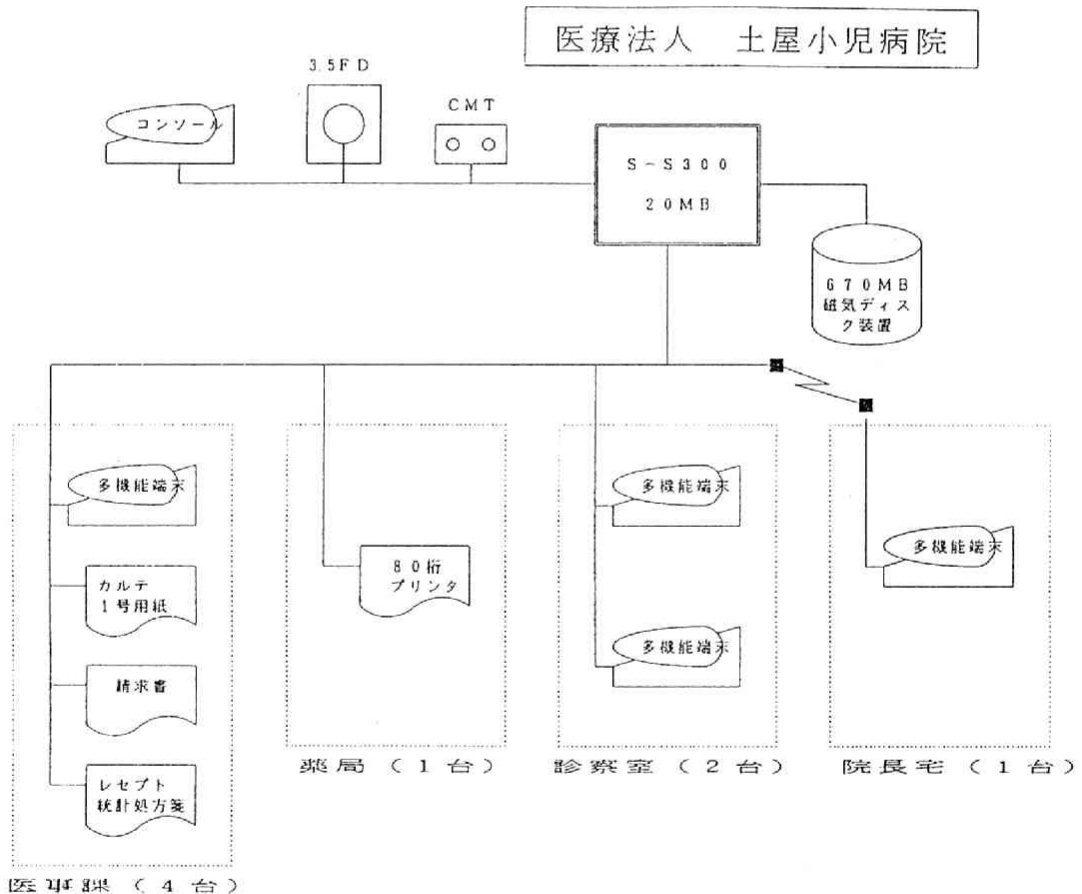
オフコンベースのシステムに比べ当院にて使用しているUNIXマシン上のパッケージシステムはまだやや不安定であり、システム管理者の居ない小病院ではさらに強固なシス

テムが必要であり、ベンダーの対応が望まれる。また、データベースのバックアップなどにUNIXコマンドの入力が要求されるため、よりユーザーフレンドリーな操作環境が望まれる。

以上、現在使用している病院情報パッケージは私共には馴染みの深いM言語を使用しているため、自作のプログラムとデータの共有が簡単に行え、実際に使用して大変便利であった。

最後に、いろいろ私共の我が儘な要求を快く引き受け、病院システム化のために貴重な意見をくださった住友電工システムエンジニアリングの皆様がこの場を借りて深謝致します。

システム構成図



千葉大学病院医療情報システム (system-CHIBA) におけるシステム運用

Utilities installed in HIS of Chiba University Hospital

- 本多正幸、里村洋一、鈴木隆弘、山崎俊司、高林克日己、
赤井ユキコ*、石井晃**、中村裕義**、野口昇**、北田光一**
清水宗***
千葉大学医学部附属病院 医療情報部 *看護部 **薬剤部
***住友電気システムズ

M.Honda, Y.Satomura, T.Suzuki, S.Yamazaki, K.Takabayashi
Y.Akai, A.Ishii, H.Nakamura, N.Noguchi, K.Kitada, H.Shimizu

千葉大学病院では、1991年より新システムに移行し、system-CHIBAと命名した。新システムでは、富士通Mシリーズに医事会計まわりを担当させ、その他の診療支援系には、SUNワークステーション(WS)を使用し、U-MUMPSをインストールした分散型システムを構築した。Mの環境では、旧システム(DSM-11)上で稼働していたシステムを継続させるとともに、住友電気のACCELを融合した総合システムを実現した。1991年9月からは、外来処方オーダーシステムおよび再来診療予約システムが稼働している。

本報告では、system-CHIBAの効率的運用、トラブルに対する迅速な対応を目的として、各種ユーティリティの整備、付加機能の組み込みを計ったので、その概要を述べる。

1. ログオン関係

診療支援系では、SUN4/370を10台、SUN4/470を2台設置し、470をサーバー(データベース)機とし、370の中の6台をクライアント(フロントエンド)機とした。(表1)ユーザーは、fep1~fep6までのどれかのWSにログオンし、srv1やdepに登録されたデータベースにアクセスする。その際、ログオン中のユーザー情報のモニターなど管理上必要なジョブや、使用中に異常中断した場合の再接続の手続きなど、日常システム管理者側が必要と思われるユーティリティを作成し、メニュー化を計った。(図1)

2. 処方箋オンラインモニターの機能強化

医師よりオーダーされた処方情報のオンラインモニターに、表2のような検索機能や警告機能を付加した。このことにより、多くの情報の簡便な操作による検索が可能となり、各種のトラブル発生時にも迅速に対応できるようになった。

3. その他

処方箋変更業務の薬剤師による代行機能の追加や、会計のない患者に対する「予約票・薬引換券」発行システム、科別来診患者統計モニター、検査データ処理モニターなど運用上必要な機能の強化を計っている。

表1 各ワークステーションの名前とその役割り

sr v 1	(SUN4/470) : データベースサーバ (患者情報、処方履歴、他)
sr v 2	(SUN4/470) : ミラーDB
f e p 1 ~ f e p 6	(SUN4/370) : ログオン用フロント (各種テーブル)
r e s	(SUN4/370) : 薬剤部業務用 (処方箋発行、処方箋修正、他)
d e p	(SUN4/370) : 中診部門業務用 (検査データ報告、他)
l a b	(SUN4/370) : 検査部業務用 (各種検査機器の制御、データ管理)
e d c	(SUN4/370) : 教育、開発用

```

LOG ON MONITOR
1 : 連続モニター
2 : バックグラウンド起動
3 : 統計
4 : LOGON table 削除 (ログオン中)
5 : 患者診療中の削除
6 : 診療中患者のリスト
7 : ログオン不許可の解除
8 : パスワードの変更
注 : ログオン処理中です (対策 : ^[MGR,SAZ] LKMAINT)

JOB # ? >
    
```

図1 ログオンモニター画面

```

LOGONモニター
30-JUN-93 10:07 AM
現在 ログオン中の人数
FEP-1 -- 12名
FEP-2 -- 11名
FEP-3 -- 23名
FEP-4 -- 17名
FEP-5 -- 10名
FEP-6 -- 15名
TOTAL 88名
1=科別医師別ログオン表
2=科別診療中患者数
1 OR 2 ? >
    
```

図2 ログオン中ユーザー数画面

表2 処方箋オンラインモニターの強化機能リスト

- オンラインモニター画面上のファンクションキーを利用した各種検索機能
(予約状況検索・医薬品マスター検索機能・過去の任意の日時に遡った画面表示機能・未取り込み処方箋検索・バックグラウンドジョブ稼働状況検索・・・)
- 多くの機能をファンクションキーに割り当てるためのロータリー形式の採用
- 薬引換え番号または患者ID番号をキーとした、処方内容履歴の検索機能
- オーダー情報の処方箋発行システムへの取り込み遅延時の警告機能

中央検査部トータルシステムの再構築

大阪市大病院中央臨床検査部 ○中尾 満、森田 寛二、藤井 厚男、東島 正満、
巽 典之
住友電工システムズ株式会社 花坂 志郎、平山 清和

1993年5月に開院した新病院は、発生源入力によるオーダーリングシステムを採用し、病院トータルシステムの導入を図った。

中央臨床検査部では、MUMPS言語を採用した従来からの部門システムを今回、新たに再構築を図った。

今回のシステムは病院システムおよび日立検体搬送システムとの情報ネットワークを完成し、生理検査や総合診療科を含むオンラインシステムを構築した。

I) 病院トータルシステムの概要

1、ハードウェア構成と通信ネットワーク

病院ホストコンピュータはM770/8を2台を配置し128MBの共有メモリーとデータベースの2重化を図り、システムのノーダウン方式を採用した。

薬剤部や放射線部などには部門システムとしてKシリーズのCPUを配置した。

中央検査部にはSUN4/670を2台配置し12台のターミナルサーバーを介して120回線を用意した。

オーダー端末及び部門システムの入出力端末には主として約800台のFMRHL2を用いた。通信回線には410MbpsのMHLINKと10MbpsのDSLINKを介して情報のネットワークを図った。

2、オーダーリングシステム

外来診療及び病棟業務は発生源入力を基本としたオーダーリングとし、検査オーダーはM770を介しリアルタイムでSUN4に依頼情報が転送される。

オーダーは生理検査・超音波検査・緊急検査を含む全ての検査室を対象とした。

II) 検査部門システムの概要

1、システムの概要

中央臨床検査部門システムの関係概念図を示した。

新システムのソフトウェアはUNIX配下でMUMPSを選択した。臨床検査部では1976年よりPDP11シリーズを用い、MUMPS言語による検査業務を行ってきた実績が有り、ソフトウェア資産や業務機能の継続を図りながら、新システムはオーダーリングシステムへの対応や検査業務の充実を目的とした。

今回採用したSUN4/670は2CPUを搭載し、1CPUのMIPS値が27MIPSの能力を持ち、従来から指摘されていたMUMPSのレスポンスの低さを大きくカバーすることが期待される。更に、2台のSUN4をルーチン系と緊急検査及びONライ

ン系の2系統に分散しCPUの負荷の軽減を図るとともにダウン時の検査業務処理への対応を考慮した。

2、検査受付処理と検体採取の準備

検査部中央採血室では、外来患者の採血及び入院患者の凝固系検査を実施している。

患者の受付はIDカードにより行い、FMR画面上に検査可能オーダーを黄色反転表示させ患者の受付処理を行う。なお、受付処理と同時に病院HOSTコンピュータに対して受付情報を転送し、医事課システムに送られる。

受付処理された情報から8種類の採血管は自動バコードラベル添付装置により準備されその他の採取ラベルは専用バーコードプリンターにより打ち出され、採取ワークシートと照合し採取容器を患者毎に用意する。

入院患者の採取容器は前日15時までに入力された検査オーダー分に対し、検査部において上記の装置セットにより用意し、各病棟には院内用自送台車を用いて搬送する。

検査依頼受付の流れを次に示した。

3、バーコードラベルの仕様

検体採取容器に張り付けられるバーコードラベルは12桁の数字をNW7フォーマットのバーコードと患者氏名や採取容器名称など13種類の情報を印字した。

このラベルに印字されたバーコードにより検査部システムや検体自動搬送システムにおける情報の確認に利用される。

4、検体到着受付処理

検査部門システムへの検体到着処理はバーコードリーダーにより行う。なお、検体検査の多くは検体自動搬送システムとの情報ネットワークにおいて自動的に検体到着処理が行われる。

III) 検体自動搬送システムとの連携

1、システムの概要

今回開発した検体自動搬送システムは検査部門システムとオンライン化し、検体検査業務に必要な多くの機能を集約することを図った。検体自動搬送システムの主な機能は次に示す。

- イ. 検体到着受付処理
- ロ. 遠心分離
- ハ. 自動分析装置用検体分注
- ニ. 血清分離分注
- ホ. 検体保存
- ヘ. 精度管理

検体自動搬送システムはHILAS(L-780/10)により一元管理されている。検体自動搬送システムの概要は次に示した。

表中の自動分析装置22台はHILASにオンラインされ、検査結果はリアルタイムでSUN4に転送される。

IV) 生理検査室におけるオンライン化

生理検査室では、今回新たに心電図解析システムであるMUSEシステムや超音波ファイリングシステムとのオンライン化を図った。

依頼情報や患者情報は検査部門システムから情報を転送し、検査結果としての数値情報や解析コメントは検査部門システムを経由して病院ホストコンピュータに転送される。

V) 検査結果の報告

検査結果は検査部門システムによる結果の最終チェックを行った後、病院HOSTコンピュータに転送される。

一方、検査部門システムにより各種報告書の発行が行われ、同時にHOSTコンピュータには最終報告として再度結果の転送を行う。

更に、病歴システムに対して磁気テープを介して報告書の患者情報をわたすことにより、シングルピッカーによるカルテの取り出し指示を行う。

まとめ

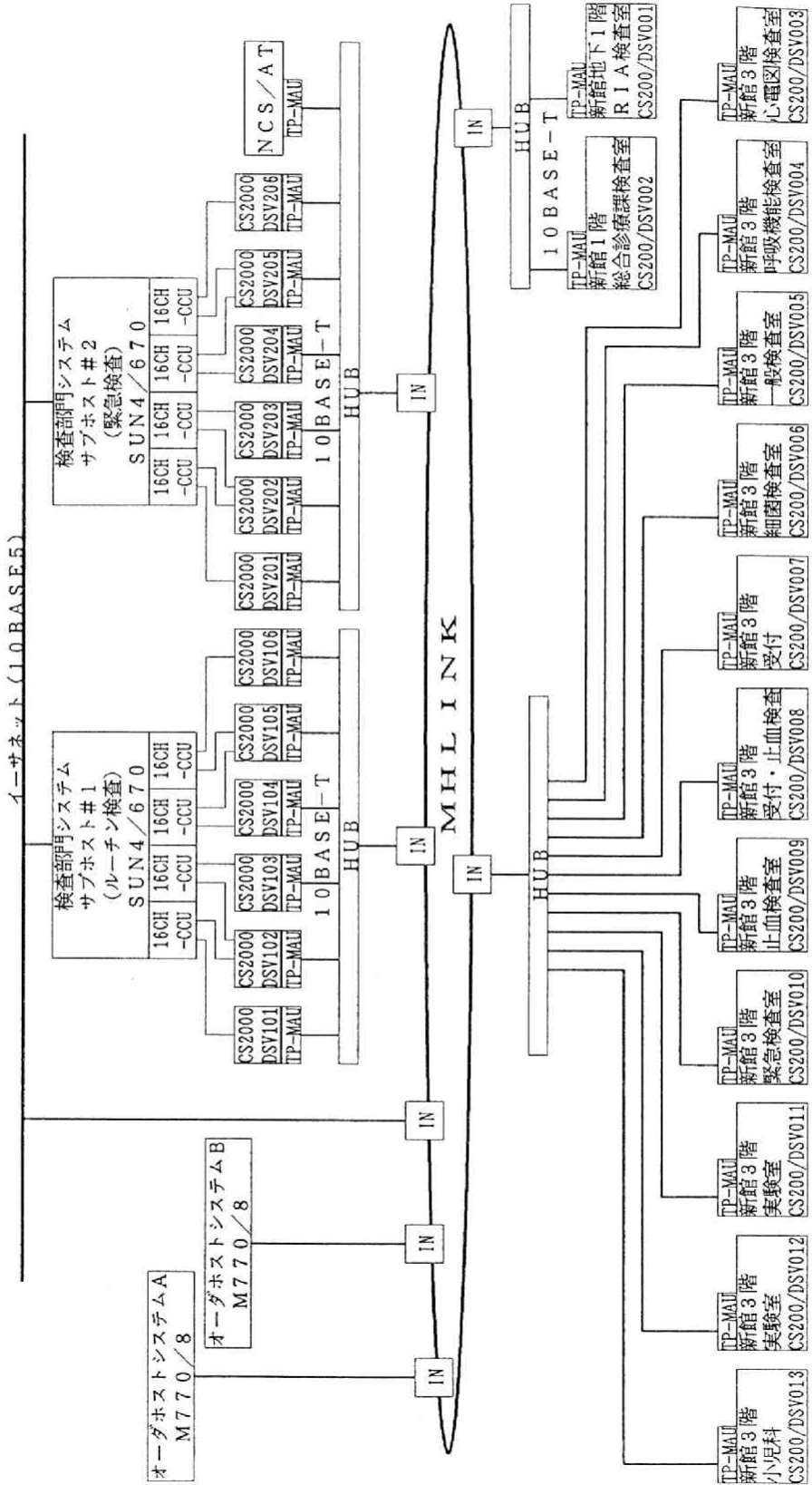
新病院立替えにあたり病院のトータルシステム化にともない、検査部門システムの再構築を行う機会を得た。

今回の再構築により検査情報の病院ネットワークが出来上がった。

中央検査部では新たな業務として総合診療科での緊急検査の支援や検査部コンピュータシステムの24時間稼働による緊急検査の実施のほか、病棟に対しては検体検査採取容器の検査部からの供給など業務の拡大を図った。

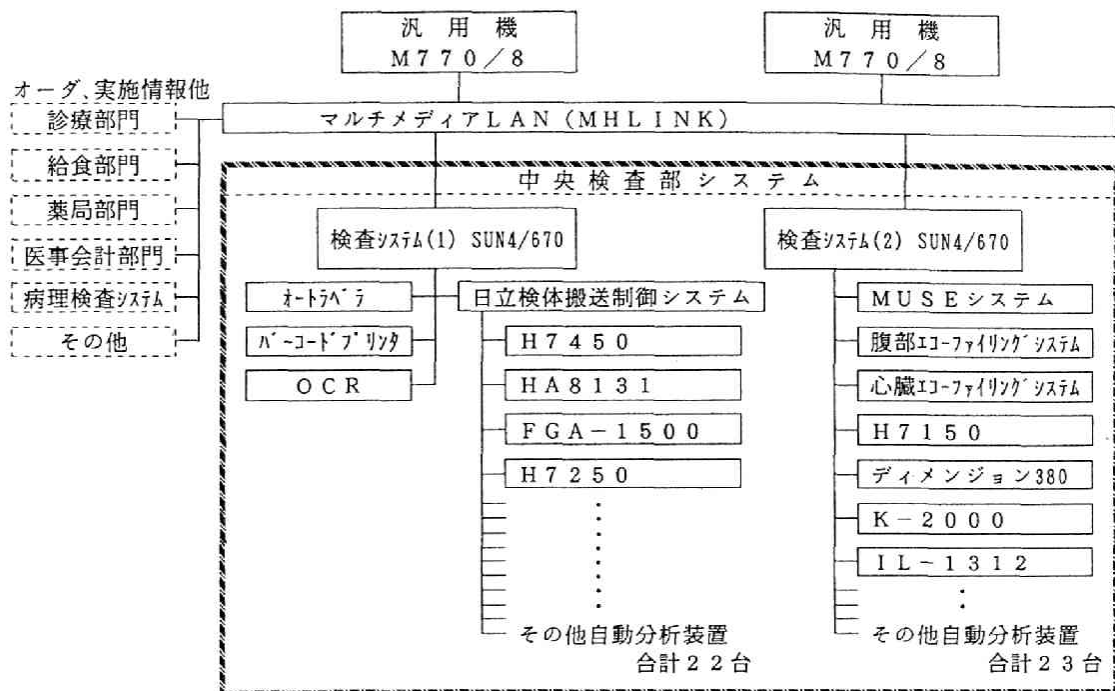
新システムの運用に入ってまだ数カ月を経過した段階ではあるが、検査部トータルシステムの初期の目的はおおむね達成されたと考えている。

大阪市立大学医学部附属病院 検査部門システム構成図

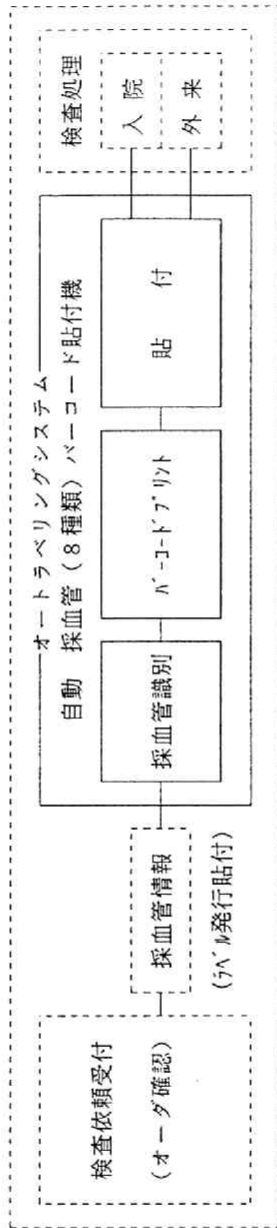


大阪市大

中央検査部システム関係概念図

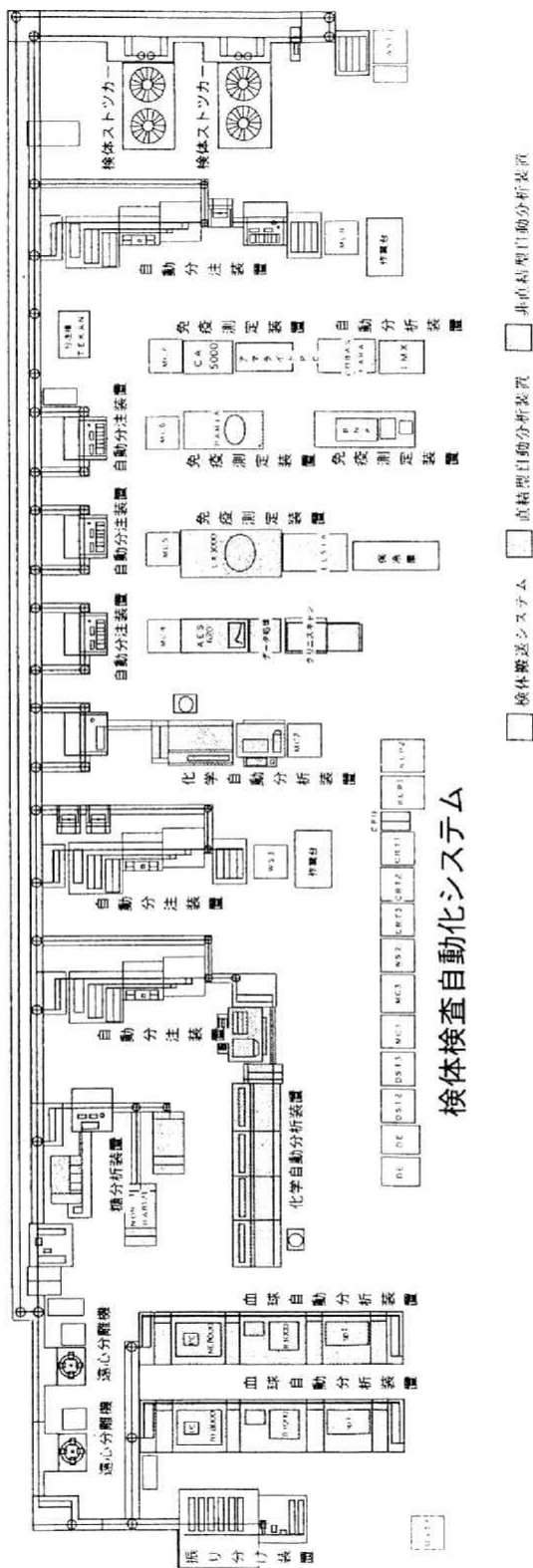


中央検査部システムに於ける受付オートラベリングシステムの概要



注) 本システムは、計3台稼働中

C. 中央検査部システムにおける検体搬送制御システムの概要



検体検査自動化システム

未熟児部および産科データベースの構築とその結合

Design and Coupling of the Two Data-base (Premature nursery and Obstetrics)

○田中吾朗^{*}, 渡辺博^{**}, 木村一元^{***}

Goro Tanaka, Hiroshi Watanabe, Kazumoto Kimura

(^{*}) 獨協医科大学第2小児科, (^{**}) 同産婦人科, (^{***}) 同総研ME室

The 2nd Department of Pediatrics, Obstetrics and Gynecology,

Division of Medical Engineering

Dokkyo University School of Medicine

1. はじめに

獨協医科大学のコンピュータ化は遅れている。学内では、個人、研究グループあるいは教室内で種々のデータベースが使用されているが、お互いのデータのやり取りは大部分が書類で行われている。ネットワーク化の希望は多いが、大学当局の関与は今のところない。MUMPSも使用されており、過去に本学術大会に多くの演題が発表されている。しかし、データベースの結合に関する研究発表はなかった。今回我々は、独自に発展してきた2つのデータベースの結合について、初歩的ではあるが、興味ある経験をしたので報告する。

2. 未熟児部および産科分娩データベース構築の経緯

未熟児部は1978年の開棟以来、現在までに3,000人を越える入院患者があり、MUMPSによるデータベースの作製は1985年である。それについては、土屋が、第12回日本MUG学術大会で「マイクロコンピュータによる未熟児データベースの作製」として報告した(1985年)。その後、開棟以来の全患者のデータを入力し、検索・集計などの使用経験を、田中が、第25回日本新生児学会で「MUMPSによる未熟児部病歴整理プログラムの作製と使用経験」として報告した(1989年)。また、産婦人科の分娩台帳は、渡辺がデータベースを1988年に作製し、それ以前のワープロの表および「日本語マイカード」のデータを取り込むことにより、1980年以後の約10,000件の分娩データを収納している。その経験を、第81回日本産婦人科学会関東連合地方会に「MUMPSによる産科分娩台帳の作製」として報告した(1991年)。

産婦人科の分娩部門と未熟児部の患者の関係は、図1の如くである。大学病院の性格から、多くのハイリスク妊婦が紹介されてくるため、新生児のうちの約20%が未熟児部に入院し、他は産婦人科に正常児ないし軽症児として入院する。一方、未熟児部の入院患者は、当院産科からの患者と院外の産科からの紹介患者により構成される。

当然、両者の間には密接な関係があり、病棟医長の間での打ち合は毎日行われており、合同カンファランスが1週おきに行われている。情報の交換は、口頭、書類のやり取り(紹介状、返書)でなされ、その度に他方のデータベースに再入力されていた。

3. データベース結合の試み

なお、使用中のパソコンはPC9801シリーズで、ソフトはSP-MUMPSである。再入力の無駄を省くためには、ホストのコンピュータにデータベースを構築し、お互いに参照すればよい訳だが、今回の発表は、その前段階としての、Floppy Diskによるデータのやり取りの話である。来年度にホスト・コンピュータとの結合を計画している。この間の発展の経緯を図示すると第2図のようになる。

結合の第1段階：相手のデータを%GSと%GRにより取り込む。

第2段階：「私のお母さんは誰？」・・・母のID番号を見つける。

Globalの第1ノードはID番号であるため、相互参照には、相手のID番号をデータとして持っていなければならない。しかし、初期の段階ではその思想がなかったため、未熟児部のデータに母親のID番号を登録していなかった。分娩台帳を見直して、母親のID番号を見つけ、再入力するには、膨大な時間がかかるため、機械的に処理することにした。

方針：母親のGlobalには、分娩日、児の出生体重、在胎週数、性がデータとして存在したので、これが全て児の誕生日、出生体重、在胎週数、性と一致すれば母親とみなして、そのID番号と姓

名を、児の Global に書き込む。

まず、母親の Global から分娩日順の INDEX FILE を作成し、児の誕生日と同じ日に分娩した妊婦の中から、一致するものを見つけることとした。しかし、機械的に処理できず、不一致のため人間の判断を要するものがしばしば見られた。

対処のプログラムの要点は以下の通りである。

W !, 児の姓名, 誕生日, 多胎の順, 出生体重, 在胎週数, 性, 紹介医, 児の ID
 S CASE=0
 D 表示
 I CASE=1 自動処理
 E 目で選択
 Q
 表示 ; 出生体重, 在胎週数, 性の一致を色を変えて分かりやすくした
 I 全て一致 X ^YELLOW S CASE=CASE+1 G DISP
 I 全て不一致 X ^RED G DISP
 E X ^BLUE ; 最低 1 つ一致
 DISP ; 分娩台帳情報
 W !, 母の姓名, 分娩日, 多胎の順, 出生体重, 在胎週数, 性, 紹介医, 母の ID
 X ^WHITE
 Q

第3段階：不一致点の修正

不一致の多くは初歩的の入力ミスであったが、特徴的な誤りの傾向もあった。

誕生日：誕生日と入院が異なる場合があり、台帳に誤記されることがあった。

この場合は、母親が見つからないことになり、別の方法で探さざるを得なくなる。院外出生児を院内出生児と誤入力していると、母親は絶対に見つからない。

体重：出生体重（産科測定）と入院時体重（小児科測定）のずれ。

週数：週数を児の身体所見から修正することがある。

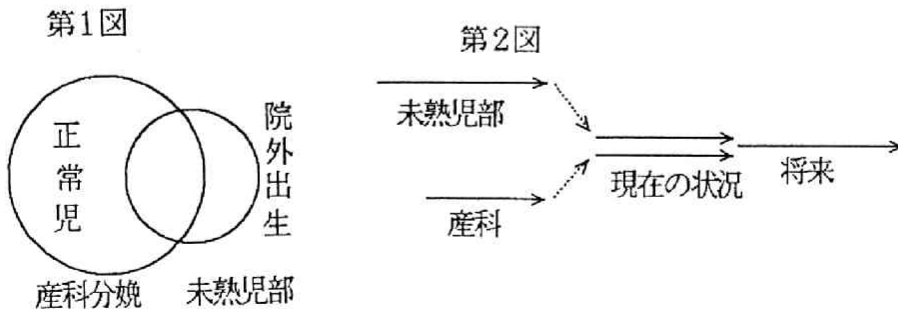
姓：これは一致の材料にできなかったが、しばしば異なっていた。

未入籍で、出産後入籍された場合、および新旧別漢字の混用が見られた。

4. 結合の成果、将来構想

結合により誤りを修正でき、データの精度が上がった。また、未熟児部のデータベースの中から母のデータを直接のぞけるようになり、一部のデータは自動的に取り込めるようになったため、入力の手間が軽減されてきた。

しかし、あくまでも Floppy Disk によるデータのやり取りであるため、相手方のデータの更新がせいぜい2週に1回位であり、その点が不満である。今回の経験が、今後の院内の他部門とのデータの共有、ネットワーク化の動きに貢献できれば、何よりであると考えている。



日本病理剖検輯報の日本語データベース

馬場謙介

国立埼玉病院臨床研究部

日本病理剖検輯報の巻末に掲載される疾病頻度表の作成過程でワークファイルが（通称、剖検輯報データベース）生成される。このワークファイルは疫学者らが剖検症例の検索に応用しているが、検索キーが疾病名に限られ、且つ、システムから直接原文が所得できないため（疾病の相互関係・ニュアンスの差等が把握できない等が原因して）、輯報の情報が十分に利用されることはなかった。そこで、1989年剖検例から輯報の印刷が電算植字化されたのを機に、電算植字の際の文書フロッピーディスクを利用することを考えた。

剖検輯報の印刷所では、数人の印刷行員が（文豪をベースとする）特殊仕様の数台のワープロで手分けして輯報の原稿を入力・編集し、複数のワープロで書き込んだフロッピーディスクを電算植字機（日本電気）に読み込ませているが、その際のプロッピーディスクのフォーマットと内容を解読して、著者のパソコン（PC-H98-70/NEC, IG光ディスク付）に移送した。移送したデータの書式を論理解析して、施設毎・症例毎に分解し、更に、（剖検番号・職業・住所・臨床診断・出所・主病変・副所見・死因・治療備考等の）項目毎に分解した後、同パソコン上のMUMPSシステム（SP-MUMPS/住電）に日本語で書かれたデータベースを構築した。設計に際し、どの項目をキー（組合せ可）にしても、症例を検索できるように設計した。検索を容易にするために、(1)用語語彙辞書、(2)リングソート機能等を付加した。検索症例の原文をプリンターに印字して提供できるようにした。

検索の実行時間は、検索論理が複雑な場合でも25分/年程度であった。検索論理の設定は、対話式に試行錯誤で進めるが、検索論理が複雑な場合でも数時間で検索論理を記述できた。検索論理の設定の方法は検索の内容によって異なり常道はないが、「簡単な症例の検索」、「脳循環障害の検索」、「胃癌で死ななかった胃癌症例の検索」等の検索の実例を示して、「検索がそう面倒でない」ことを示したい。

「健診システムにおける病名と家族歴の登録サブシステムの開発」

Development of Diseases Registration on the Health Examination System

○大榎陽一^{*}、笹川紀夫^{**}、高橋正宏^{***}、原寿夫^{****}

Yoichi Ogushi, Masahiro Takahashi and Hisao Hara

* 東海大学医学部医学情報部、** 島根医科大学、*** 郡山市健康振興財団、
**** 郡山市医師会

* Division of Medical Information, School of Medicine Tokai University,
** Department of Medical Informatics, Shimane Medical School
*** Koriyama-shi Health Promotion Foundation,
**** Koriyama Medical Association

1. はじめに

我々は、先に人口約30万人の郡山市を対象とした地域健康情報システムの構想とその最初の事業である事業所健診システムの開発について報告した。このシステムは一日60人の健診データの処理を行っている。ハードウェアとしてはデスクトップ型パーソナルコンピュータをホストとし、4台のノートブックパソコンとオフラインネットワークにより構成されている。アプリケーションはSP-MUMPSを使い開発されている。今回、このシステムに受診者本人の病歴と家族歴を管理するサブシステムを開発したので報告する。

2. システムの概要

ホストはPC98-RL（内臓ディスク：40MB）+外付けハードディスク300MBである。これにはバックアップ用のストリーマテープ（150MB）と報告書や統計類の出力用のプリンターが接続されている。端末としてノートブックパソコン（ハードディスク内臓）が4台導入された。ホストと端末間はフロッピーディスクでデータをやりとりすることにした。オフラインネットワークとしている理由は次の通りである。①健診の受付登録から報告書の出力までの日常業務が一本の流れになっており、データベースの非同期な更新が無い。②オンラインネットワークに比べて経済的負担が少ない、運用管理的の専門家が不要である。③オフラインネットワークの方が行動範囲が制限されない。しかし、オンラインネットワークに対して劣る点も存在するため次のような工夫も行い、使い易さの改善とデータベースの安全性の確保を行った。①端末には指定する受診者の過去分を含むすべてのデータを移せるようにした。②端末側のアプリケーションは、集計処理などを除き、ホストと同じとした。③従って、端末での受診者登録や健診結果の登録などが分散入力出来るようになった。④データベースの消去などの処理のミスを防ぐため、アプリケーションでホストと端末の自動識別を行った。なお、検査システムからも従来どおりフロッピーにてデータを受け取っている。

稼働している業務を次に示す。*は今回追加された項目である。

①個人基本情報（登録、照会、修正、削除）

個人コード、地区コード、世帯番号、セイメイ、姓名、生年月日、性別、住所、電話番号、事業所コード、第二個人コード（事業所内ID）、病歴^{*}、家族歴^{*}

②個人コード照会（個人基本情報から、日付より、最近使用コード）

③事業所健診録（登録、照会、修正）

身長、体重、尿検査、血圧、視力、聴力、胸部X線、心電図、眼底、診察所見、指導区分、メモ

④健康増進メディカルチェック（登録、照会、修正）

（事業所健診と同様）

- ⑤ 検査結果取り込み
検査システムのフロッピー入力
- ⑥ ガン検診結果（登録、照会、修正）
胃癌、肺癌、大腸癌、乳癌、子宮頸癌、子宮体癌
- ⑦ 各種事業実施（登録、照会、削除）
各種教室、各種相談など
- ⑧ ホスト・端末間データ転送
ホスト→端末出力、ホスト→端末入力、端末→ホスト出力、端末→ホスト入力
- ⑨ グラフ表示とコピー出力
時系列表示、レーダーチャート
- ⑩ テーブル（登録、修正）
地区、事業所、項目、事業名称、肥満度判定、グラフ表示パラメータ
- ⑪ 帳票
個人通知書、一覧表（第二個人コード順）、要医療者リスト、ガン検診結果一覧表、集計表（男女別・年齢別・臓器別・判定区分別）
- ⑫ 個人コード変更
関連データの一括移動を含む

3. 病名マスターの作成

健診システムであり、確定診断を行った上での病名では無いので、独自の簡易病名マスターを作ることも検討されたが、医療との連携や今後の地域健康システムとしての発展を考慮し、医療情報開発センター（MEDIS-DC）の病名マスターを購入し、M言語のグローバルを初期セットした。この構造は次の通りである。

^ZTDSMST(BYOMECD)

キー① MEDIS病名コード（約28,000）

データ 漢字病名；カナ病名

病名コードの検索用のインデックス・ファイルの構造は次の通りである。

^ZTDSINDEX(KEY,BYOMECD)

キー① 検索用カナ文字列 キー② MEDIS病名コード

データ null

この初期設定はカナ病名から自動的に行った。この手順は次の通りである。

- (1) カナ病名に含まれるスペース文字、<、>、[、]、.、(、)、/により文字列を分割する。
- (2) 分割された文字列と、それらの結合による文字列を作る。
- (3) 各文字列について、検索文字列変換を行う。検索文字列変換とは、小文字を大文字にする、バンクチュエーション文字を除去する、デとジなどの日本語の同一発音文字列を統一する、ウィとビなどの外来語の類似発音文字列を統一する。なお、この処理は実際に検索文字列が入力された場合も行い、インデックスを検索する。

病名コードとインデックスの追加を適時行えるように、病名テーブルメンテナンス・ルーチンを作成した。これらのプログラムでは、登録、修正、削除、画面での一覧表示、リスト出力が行える。

4. 登録の方法

個人基本情報登録画面の「f・2」キーにより本人の病歴登録画面へ、「f・3」キーにより家族歴の登録画面に遷移する。本人の病歴としては病名、開始および終了年齢、経過、メモが登録される。これらは一画面に5組表示されるが、ページングにより何組でも登録が可能である。病名選択では、コード入力、頭文字入力、頭文字のローマ字変換入力の3通りの方法がある。家族歴としては、本人との関係、病名を1画面当たり10組表示できる。ページングが可能で、登録数の制限が無いことは同様である。また、病名の選択方法も全く同じである。

5. おわりに

この病名登録システムは、健診により要医療となったものについてのフォローアップシステムや、各種統計などに使い、健診受診者本人の健康管理のより一層の支援や、職場や地域における健康上の問題の発見と解決につながることを期待されている。

【参考文献】

1. 大櫛陽一他：地域健康情報システムとしての事業所健診システムの開発。第18回日本MUMPS学会大会論文集、99-106、1991.
2. 大櫛陽一：医療情報システムにおける最近のトピックスー病院情報システムから地域健康情報システムへ。あいみっく、12(2)、4-10、1991.
3. 大櫛陽一：これからの医療に貢献する情報技術。新医療、18(7)、42-46、1991.
4. 大櫛陽一他：ダウンサイジング化された健診システム。第12回医療情報学連合大会論文集、67-68、1992.

「選択必修科目履修申告システムの開発」

Development of Registration System for Elective Courses

○齊藤 成広、松木 三徳、菊地 良平、大楠 陽一*

Shigehiro Saito, Mitsunori Matsuki, Ryohei Kikuti, Yoichi Ogushi*

東海大学伊勢原学務課

*東海大学医学部医学情報部

The Educational Affairs Section at Isehara, Tokai University School of Medicine.

*The Department of Medical Information, Tokai University School of Medicine.

1. はじめに

東海大学医学部では1988年度より、いままでの画一的な教育を改め、新しい医学教育を目指してCOS (Case Oriented System) カリキュラムによる教育を実施した。このカリキュラムの主な特色は、①基礎・臨床統合カリキュラム、②スモールグループによるケーススタディ、③選択制カリキュラム、④医学英語・交換留学制度、⑤チュータ制度、⑥コンピュータ教育の導入等であり、本年度に完成年度を迎えた。

その特色の一つである「選択制カリキュラム」は3年次から6年次にかけて実施され、その制度のもとに開講される選択必修科目は約400科目、全授業時間数の1/3にもわたり、毎週月曜日と木曜日の午後に開講され、かつ、6年次は教育課程の全体が選択制となり、フルタイム選択必修科目として開講される。

この選択制カリキュラムの特徴は、自分が将来の希望や目標に向けて履修計画を組み、それに従って学習できることにある。しかし、現実には各科目における受入れ人数、年間開講回数、履修条件等による制約のため、受入れ人数を越えた場合は単純に大型コンピュータによる無作為抽選を行い、履修者を決定していた。そのため、必ずしも学生が満足できるような履修計画にはならなかった。さらに、学生は希望の科目が無作為抽選により漏れた場合のことを想定して、単位数(時間数)を稼ぐために多めに科目を申告し、本当に受講したい学生が抽選に漏れてしまうという問題が発生した。このことは選択制カリキュラムを検討した段階で、起こり得る問題として予測できたことではあったが、本学の全学的教務事務システムになじまないことや、独自のシステム構築には、財政的、労力的リスクが大きい等の理由で課題解決が遅れていた。

そこで、一人一人の学生の希望に沿ったカリキュラムが組めるという、日本唯一の教育システムを生かすために、希望履修科目に優先順位をつけた、新しいシステムを開発する必要があった。このことにより、標記のシステムをMUMPSを用いて開発することになった。

2. システムの概要

本システムは、病院システムとは別個にPC-9801上でSP-MUMPSを用いて開発された。ただし、開発にあたっては最終的に病院システムの中に組込まれ、運用されることを前提に共通の文法、ツールにより開発している。

このシステムは、履修申告科目ごとに希望履修優先順位がつくため、新しいマークシートの作成作業も同時に行った。履修申告の方法は、学生が履修申告票(マークシート)を提出後、ただちにマークリーダーで読みとり、受付窓口で「履修申告確認表」を出力し、その場でエラー修正が出来るようにした。また、履修申告期間を約1ヵ月半設け、その間に週1度、

「履修申告科目別履修人数一覧表(ワズ表)」を発表し、何度でも履修申告のやり直しができるようにした。つまり、学生間の自主調整を行わせたのである。それでも履修希望人数が受入れ人数を越えている場合は、本システムによる優先順位ごとのコンピュータ抽選を最後に1回のみ行うことにした。抽選方法は、各学生の優先順位第1位の科目をまず最初に対象とし、受入れ人数を越えている科目のみ乱数を発生させ、抽選を行い、履修者を決定する。もし、この抽選に漏れた場合は、その学生の第2位以降の科目の順位を1つづ繰上げて、次順位以降の優先権を与えるようにした。そして、第1位の順位が無くなるまで繰返し、第1位が無くなると第2位の順位へ移行していくという方法である。また、申告科目が既に人数が埋まっている場合にも、次順位以降の科目の順位を1つ繰上げるようにした。このような方法で、全学生の最終順位が無くなるまで繰返す、コンピュータ抽選の方法を考案したのである。

3. 結果

開発は10月より開始し、本システム仕様の決定まで、通常の業務を持ちながらの開発となったが、10・11月に機能仕様、外部仕様、プログラム設計、ファイル設計等を大槪と打ち合せしながら決定し、12月、翌1月まで大槪と初心者である筆者と二人でプログラムの作成、テーブルの登録、テスト運用等を行い、2月より稼働した。約4ヵ月間の急な開発であった。

その結果、システムを動かすプログラムは91本、ベースとなる基礎データのテーブルは9つ、データ等を元にして作られるファイルは3つ、ディスク容量は20MBを使用している。

本システムで扱う主な画面としては、履修状況の照会・検索、データ更新、リスト、データ交換、テーブル更新に大きく分け、サブメニュー、サブメニューにより学生個人情報、教員情報、科目情報、テーブル情報と細部の情報が画面で確認でき、かつリスト出力も出来る。

マークシートの読み込みは、1日数枚から250枚前後とばらつきが多かったが、約1ヵ月半の履修申告期間中に、システム上のトラブルは特に発生しなかった。

履修申告科目別履修人数一覧表(ワズ表)は、計7回公表したが、最終ワズ表公表後に学生の履修申告の修正が集中し、当初、考えていた学生間の自主調整は今一つの感であった。

その後、前述のコンピュータ抽選を実施した。その結果、履修科目確定後の学生個人間の履修確定単位数(時間数)のばらつきが大きく、一部学生からはワズ表で変更した申告内容が無駄であったり、裏目と出たとか、コンピュータ抽選後の申告追加を実施して欲しい等の要望が出された。この単位数のばらつきの大きな原因は、昨年度までとの申告方法の相違による学生間の理解度の低さやとまどい、大学側の説明の不十分さによるものではないかと考えられる。

事務処理側からは、特に使用初年度でもあり、システムに関する理解度の低さから来る誤操作があった程度で、大きな問題は発生しなかった。

4. おわりに

初めてこのシステムを使用した本年度は、一応の成果をみた。以前のやり方に比べると希望の科目がすべてとは行かないまでも、学生の希望に沿った履修ができるようになったからである。今後の予定としては、前述の学生からの要望を取り入れ、申告の追加、抽選の複数実施を可能にすべく、関係プログラムの修正を行い、急ピッチの開発のため行届かなかったシステム全体の操作性のレベルアップを行うと共に、各テーブル間のデータチェック機能を加えることにしている。

SQLによるDSMデータベースへのアクセス

Access to DSM database using SQL

辰己岳欣

Takeyoshi Tatsumi

日本ディジタルイクイップメント(株)
西日本第一統合システム部

Digital Equipment Corporation Japan
Enterprise Integration Center

〒530 大阪市中中之島2丁目2番2号
ニチメンビル

TEL(06)-222-9211 FAX(06)-202-1419

要旨

SQLを用いてDSMデータベースの検索や更新を行なうためのインターフェースドライバを開発中です。これによって既存のRDB検索ツールからDSMデータベースをアクセスしたり、また、クライアント/サーバアプリケーションの構築が容易に行なえます。

We have been developing an interface driver with SQL to retrieve and update DSM database. Using this driver enables us both to access DSM database by way of public retrieval tools for RDB and to easily develop client/server application.

キーワード: DSM DASL SQL RDB Client/Server

はじめに

弊社では、RDB for OpenVMSやリレーショナルデータベース以外のデータベースに対して、弊社標準のDigital Standard Relational Interfaceを用いたSQL(Structured Query Language)アクセスを行なうための一連の製品群を計画しています。この計画に基づき、RDBaccess for RMS, RDBaccess for ORACLE等が現在製品化されています。標準のSQLコマンドによるDSMへの検索、更新を可能にする、「RDBaccess for DSM(仮称)」はこの製品群の一部として位置付けられます。RDBaccess for DSMは、

† ORACLEは米国Oracle社の商標です。

RDBの世界で提供される標準SQLツールをDSMデータベースに対して使用することを可能にします。

RDBaccess for DSMの概要

RDBaccess for DSMは、テーブルのメタ情報を管理する「メタデータドライバ」と実際にデータアクセスを行なう「データドライバ」からなります。DASL(DSM application software library)検索ドライバを用いてエミュレートされたRDBテーブルから、メタデータドライバが、フィールド長、フィールドタイプなどの情報を取得し、それに基づいてデータドライバが実際にグローバルからデータをRDBレコードとして検索、更新します。

必要なデータの要求は、これらのドライバの上位レベルにあるNon SQL Database Serverが行ないます。図1は、データの流れを示しています。

クライアント/サーバモデルの構築

Digital Standard Relational Interfaceを用いたものとしては、通常のSQLコマンドインターフェースの他に、クライアント/サーバアプリケーション構築用のAPIを含むSQL servicesがあります。SQL servicesを用いれば、DSMをサーバデータベースとするクライアント/サーバ型のアプリケーションが構築できる他、クライアント側で用意されている既存のSQL services対応のソフトウェアが利用できるようになります。

また、将来的には、米国Microsoft社がデータベースアクセスの標準インターフェースとして提唱しているODBCドライバを提供する予定ですので、図2に示されるように、ODBCに準拠したMS-Access、Visual Basic等のクライアントツールからDSMデータベースにアクセスできるようになります。

まとめ

RDBaccess for DSMを用いれば、エンドユーザは、データベースの種類(RDB,DSM,...)を意識すること

† MS-Accessは米国Microsoft社の商標です。

† Visual Basicは米国Microsoft社の商標です。

なくデータ処理が行なえます。また、既存の RDB 用の検索ツールや開発ツールが使用できるので、アプリケーション開発の幅が飛躍的に拡大します。

図 1
データの流れ

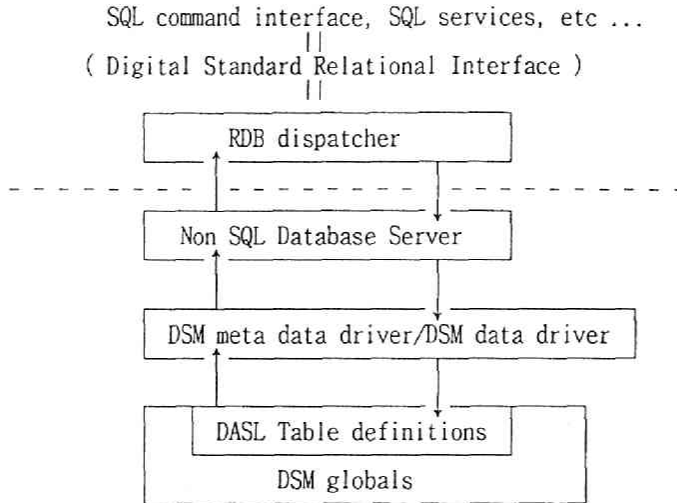
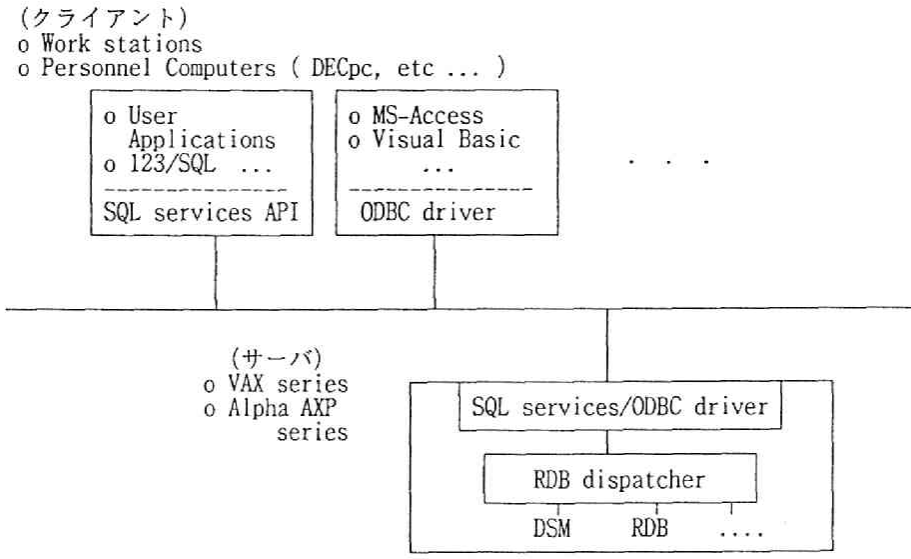


図 2
クライアント/サーバモデル



U-MUMPS クロスシステムジャーナルによるミラーシステムの構築
Construction of mirror system with U-MUMPS Cross System Journaling

○吉村 貴由, 木村 隆, 伊藤 徹, 吉中 位知朗, 藤江 昭
○Kiyoshi Yoshimura, Takashi Kimura, Touru Itou, Ichirou Yoshinaka,
Akira Fujie

住友電気システムズ(株) 応用システム事業部*1
Sumitomo Electric Systems Co.,Ltd.

抄録

我々はU-MUMPSの障害対策機能として、ネットワーク上のワークステーションにミラーデータベースを構築するクロスシステムジャーナルを開発した。クロスシステムジャーナルにより、障害時におけるシステムの停止時間を大幅に短縮することができる。

ABSTRACT

We developed Cross System Journal by which the mirror database was constructed in the workstation in the network as a trouble measures functions of U-MUMPS. The stop time of the system at the trouble can be greatly shortened by Cross System Journaling.

Keywords

U-MUMPS、障害対策、ジャーナル、ネットワーク
U-MUMPS, trouble measures, Journal, network

1 はじめに

U-MUMPSは、unixワークステーション上で動作するマルチユーザ日本語対応MUMPSである。我々は1984年より、U-MUMPSをベースとする医療情報システムを構築、販売してきた。近年の医療情報システムは部門システムから総合システムへと成長し、現在では病院業務のバックボーンに位置づけられる。従ってシステムの障害は病院業務に多大な影響を与えることになる。従来のU-MUMPSにはシステムの障害対策としてジャーナル機能(アフター・イメージ・ジャーナル)が用意されていた。しかしジャーナルはデータベースの保護機能であり、CPU障害等の重要障害に対し、システムは長時間に渡る停止を余儀なくされる。我々はU-MUMPSの障害対策として新たにクロスシステムジャーナル機能を開発し、障害時におけるシステムの停止時間を短縮することに成功した。

2 クロスシステムジャーナルの概要

U-MUMPSクロスシステムジャーナルは、イーサネットで接続された2台のunixワークステーションから構成される。

1台のシステムを業務システムが稼働するプライマリシステムとして機能させる。プライマリシステムでは、プライマリU-MUMPSがデータベース更新に対し、プライマリ、ミラー両システムのディスク上にジャーナルファイルを作成する。更にミラーシステムにジャーナル情報を記録したコントロールファイルを作成する。ネットワーク通信にはTCP/IPプロトコルを使用する。

もう1台のシステムはプライマリシステムに対するミラーシステムとして機能させる。ミラーシステムでは、ミラーU-MUMPSがコントロールファイルに書かれたジャーナル情報に従い、プライマリU-MUMPSが作成したジャーナルファイルを順次読み込み、ジャーナルデータをデータベースにリストア(デジャーナル)する。

これによりミラーシステムはプライマリシステムに対するホットスタンバイとして機能する。

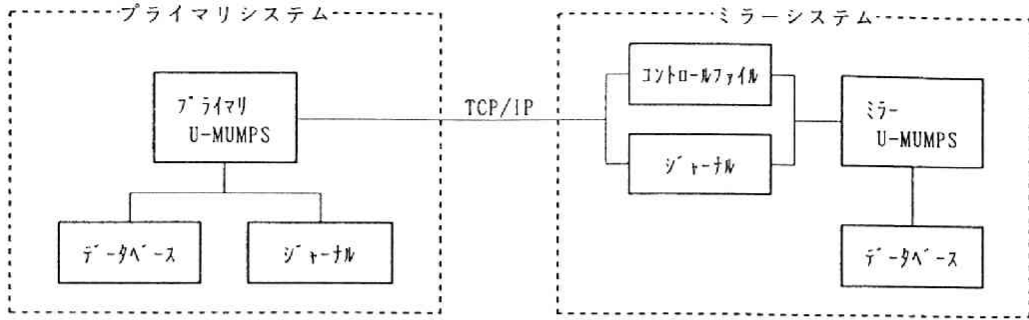


図-1 クロスシステムジャーナルの概要

クロスシステムジャーナルにより、業務システム（プライマリシステム）に障害が発生した場合、業務システムをミラーシステムに切り換えて直ちにシステムを再開することができる。

ジャーナルデータをファイル渡しにすることで両システムの機能は完全に独立している。このため業務システム稼働中においても、ミラーU-MUMPSを停止させデータベースのバックアップを取ることができる。バックアップはACCCELバックアップキットにより、自動バックアップが可能である。

更に未送信ファイル再送機能により、システムの起動順序に左右されない柔軟な運用を可能にしている。

3 処理性能

プライマリU-MUMPSは、ジャーナルデータを8Kバイト単位にネットワーク送信するため、従来のジャーナル機能と同等の処理性能を実現している。バッファ中のジャーナルデータは指定時間ごとにフラッシュし、障害時のデータ保障範囲を明確にしている。

ミラーU-MUMPSは、プライマリ側より処理量が多いためボトルネックが予想されたが、既存システムの実データによるシミュレーションテストの結果、最大トランザクションの3倍の負荷でもミラー側の処理が遅れないことを検証している。

4 ミラーシステムの構築

クロスシステムジャーナルにより、部門システムまたはDDPによるファイルサーバーを2重化することができ、システム全体のフォールトトレランスを高めることができる。

ミラーシステムの構築例を図-2に示す。

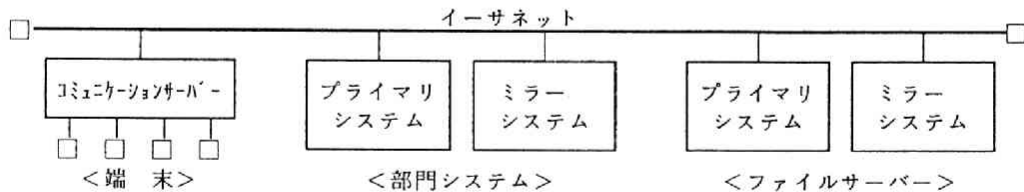


図-2 ミラーシステムの構築

5 おわりに

当初の目的に加え、従来のジャーナル機能と同等の性能を実現できたことは大きな成果である。しかし、障害時のシステム切り換えに人的操作を必要とすること、障害時に不整トランザクションが発生する場合があるなどいくつかの課題を残している。これらの課題を含め更なる機能拡張を行い、信頼性の高いシステムを目指してU-MUMPSの開発を進めていく考えである。

M と TP モニターとの統合の試み

An Approach to Integrating M with a TP monitor

○今井敏雄、 佐藤真美

○Toshio Imai, Mami Sato

日本デジタルイクイップメント(株)
西日本第一統合システム部

Digital Equipment Corporation Japan
Enterprise Integration Center

〒530 大阪市中中之島2丁目2番2号
ニチメンビル

TEL(06)-222-9211 FAX(06)-202-1419

While trying to integrate M into systems and tools outside, we will probably face some interfacing problems such as data-type conversion and how to meet the dynamic property of M with static one of compiler's language. This article describes a 4GL approach we adopted as a practical solution to the problems. The target for integration in our case is an OnLine Transaction Processing Monitor called DEC ACMS(Application Control and Management System). When a certain development project was in its design phase, we chose to employ DSM and DSM application Software Library(DASL) as a front-end or presentation part of the system based on ACMS. We, therefore, needed and developed integration tools for ACMS and DSM, which we call "DASL for ACMS". We describe its outline in this paper.

キーワード：TP モニター,4GL,DSM,DASL,ACMS

はじめに

26年も前に誕生したMUMPSの実用性、使いやすさ、性能は、現在でもその魅力は十分あります。しかし、高性能PCの普及を契機とするコンピュータユーザー層の拡大にともない、GUIを駆使した使いやすくて、高機能なものが世の中に出てきており、以前のような「MUMPSでなければならないんだ」という差別化された魅力と迫力が薄らいできていることも

事実だと思っています。Mが、今後も発展し続けるためには、M自身の魅力を高める努力と同時に、M以外の世界との関係を深め、その共存の中で生きていける力がMになければならないと思います。

MとM以外の世界との結合時に、Mで問題となるのは、以下の3点です。

1. タイプフリー

Mは文字列型のデータタイプのみで、データ定義のメタデータを持たない。そのため外の世界のデータタイプ/コードとの変換をどこでどう行うか?

2. 実行時評価

コンパイル型言語との統合時、そのスタティックな要素とMのダイナミックな要素をどう折り合いをつけるか?

3. シンプルフロー

GUIのイベント割り込みや、通信等での並行実行の非同期処理をどうハンドリングするか?

これらは、Mの世界の中だけで考える場合、Mの使い易さの源泉であり、大いなる特徴です。しかし、他の言語やシステムとデータや処理を分担し、統合利用する場合、逆に障害となります。この障害をなんらかの方法で、解決する努力をM側で行なわないとMは孤立化する恐れがあります。これに対し、MUMPS Development Committee(MDC)で議論されている成果(SQL, Windows, RPC, C等のインデングやObject MUMPS)に大いに期待する所ですが、ジェネレータタイプの4GLアプローチで、この問題に対し、現実的な解決を試みた例がありますので、以下に報告します。

DSMとの統合の対象の外の世界は、弊社トランザクション処理システムのACMSです。

TPシステムとM

コンピュータシステムが社会の中核部分で利用されるようになればなるほど、より高性能で、かつ、信頼性の高いシステムが求められます。オンライントランザクション処理(OLTP)システムがそのひとつの解答です。弊社のApplication Control and Management System(ACMS)は、そのTPシステムです。

TPシステムは、通例、MのTime Sharing System(TSS)ベースのシステムとは全く異なる実行環境をもちます。それは、いわゆる、Client/Server環境、あるいは、Front-end/Back-end分散環境です。Mは、データベースアクセス手続も言語内部に取り込むという特異なアプローチで、プログラム言語とデータベースを一体化させ、さらに、端末アクセスも含めて、すべてをコンパクトに一体化させることで、最適化と使いやすさを図ってきました。これとは逆の方向、つまり、プログラム言語とデータベース処理を完全に分離するアプローチで最適化を図りAPIを標準化してきたのがRDB/SQLです。そのRDBをベースにするTPシステム(ACMS)は、更に、分離要素化アプローチを進めて、フロントエンドの端末処理部、バックエンドのデータベース処理/手続処理部、そして、それら全体を管理し、多端末/多トランザクションを効率よく処理するTPモニター部に機能分割しています。

このモジュール構造は、それぞれの要素の機能特化が図れると共に、中枢モニター部がトランザクションと構成管理を行いますので、システムの高可用性(対障害性)、柔軟性と拡張性、データベースの一貫性とセキュリティ保持、協調分散処理等の達成に有効なアプローチです。しかし、Mのようにシンプルでないことは事実です。簡便コンパクトなMと、重厚モジュール型のTPシステムとの文化/発想の差を感じさせます。Mは、その簡便さ(特に開発/保守性と性能)のメリットを保ちながら、TP処理や、Client/Server環境にどう適応/進化すべきかが、ひとつの大きな課題です。

今回、ある開発プロジェクトで、ACMSの端末処理パートシステムとしてDSMとDSM application Software Library(DASL)を使用するため、ACMSとDSMとの統合ツールを開発しました。

ツール名称を「DASL for ACMS」と呼びます。

目的は、Mの特徴を生かし、端末/フロントエンド部の開発/保守生産性の向上と、TPシステムのフロント処理効率アップで、MがTPシステムのクライアントとなる試みです。

ACMSとAPI

ACMSは、タスク言語で定義した「タスク」の実行を行う「タスクサーバ」を制御下に、データベースアクセスを行う「プロセッシングサーバ」、端末処理を行う「プレゼンテーションサーバ」、および、タスクを呼び出す「クライアントプログラム」から構成されます。「クライアントプログラム」のひとつにユーザ作成のプログラムからタスクを呼ぶAPIであるCustomer-Written Client Program(CWCP)があります。CWCPを利用して、DSMとの結合を行います。DSM側の接点は、M標準規格化される外部ルーチン呼び出し External routine calling syntax (E

ALL)を利用します。EALLインタフェースプログラムはCで書きます。

ACMSの実装は、ポータブルなClient/Serverモデルを実現するため、Distributed Computing Environment(DCE)のRemote Procedure Call(RPC)をベースにしています。ACMSタスク言語コンパイラは、クライアントに対しては、対象タスクの引き数レコード定義を含むCヘッダーとクライアントスタブオブジェクトを生成します。クライアント側のプログラムであるDSMは、生成クライアントオブジェクトとのリンクを行います。

Mとの統合のために、インタフェース部で、タスクコール引き数レコード構造からフィールド切り出し/合成を行う処理、フィールド値のACMSサポートデータタイプ(OCTET, INTEGER, TEXT, NATIONAL TEXT, ARRAY, RECORD)とM変数(可変長文字列)の変換処理、外部(漢字)コード系と内部(データベース)コード系との変換処理等を行わなければならない。これらの処理を、アプリケーションタスク毎に、インタフェースCプログラムで書く場合、その数が多くなるとインタフェース部の開発、保守の負担が大きくなり、統合のメリットがなくなります。ここがMと他の世界との統合上、問題となるところで、Mがその世界に閉鎖的となる傾向の技術的素因だと考えます。この問題に対し、4GL(第4世代言語)アプローチでこの解決を試みました。

4GLアプローチ

ACMSとの統合で、採用した方式は以下のとおりです。

- ① M側に、外部統合に必要なメタ定義情報を取得保持する。
- ② ①の外部メタデータをM側のメタデータにどう対応づけるかのマッピング情報を持つ。
- ③ ①②をもとに、Mから外部呼び出しをするMDライブラリーチェーンおよびインタフェースCプログラムを自動生成する(4GL)。

このことにより、Mアプリケーションは、生成ルーチンと呼ぶだけで、あたかもM中のサブルーチン呼び出しのように外部アクセスが出来るようになります。外部世界のレコード定義分解/合成、データタイプ/コードの変換、外部エンティティデータとMエンティティデータのマッピング、外部呼び出しシーケンス/エラー等の統合にかかわる処理は、生成ドライバールーチンがハンドリングしますので、その詳細をアプリケーションから隠蔽し、アプリケーションの抽象度を高めることができます。また、アプリケーションとEALLおよび外部の間に、ドライバー緩衝層を設けますので、結合下層/環境の変化、技術進歩に対しても、4GLの進化で柔軟に対応できます。

ACMSとの統合で、上記アプローチを、DASLのオプション拡張として実現します。DASLは、DSM上の4GLで、スクリーン/レポート/検索のDSMルーチンを生成します。また、DSMの標準開発実行環境を提供します。DASLには、グローバルデータ構造をはじめ各種メタデータ定義辞書を持っています。外部世界の定義情報をその拡張として保持することが自然と考えました。

DASL for ACMSの概要

ACMSの端末プレゼンテーションパートとして、DASL/DSMを有効に使うため、ACMSとDASLとの間のインタフェースプログラミングの開発生産性/保守性/品質を上げることを目的とするツールです。

DSMルーチン、DASLアクション部から、ACMSへのアクセス(タスクコール)が、サブルーチン呼び出し、

```
DO CALL^%DAACMS(group,task)
;group=タスクグループ名、task=タスク名
```

でアクセスできるようにします。以下の3つの機能が、主機能です。

① タスク定義取り込み

ACMSタスク言語アプリケーション ソースプログラムからタスク呼び出し情報(レコード定義、タスク引き数定義)を抽出し、DASLのタスク定義辞書データベースとして保持する機能。

② タスクコール定義

ACMSタスクコール引き数レコードの各フィールドを、DASLのデータネーム(エンティティ名)やスクロールアレイ、DSM変数にどうマッピング(対応づけ)するかを登録する機能。このマッピング情報は、ACMS側とDASL側のインタフェーステーブルの役割をします。この仲介マップ情報を持つことにより、M側アプリケーションでは、DASL/DSM内の変数でタスクコールのデータを引き渡すことができ、ACMSとDASLとのエンティティ名の独立性と、関係の柔軟性を保証します。

③ タスクコールコンパイラ

①,②の情報をもとに、

```
DO CALL^%DAACMS(group,task)
```

で呼ばれる、ACMSタスクコールインタフェース(DSM)ルーチンを生成する機能。タスク定義、マッピング情報の変更に対しても再コンパイル処理で、インタフェースルーチンを再生成でき、イ

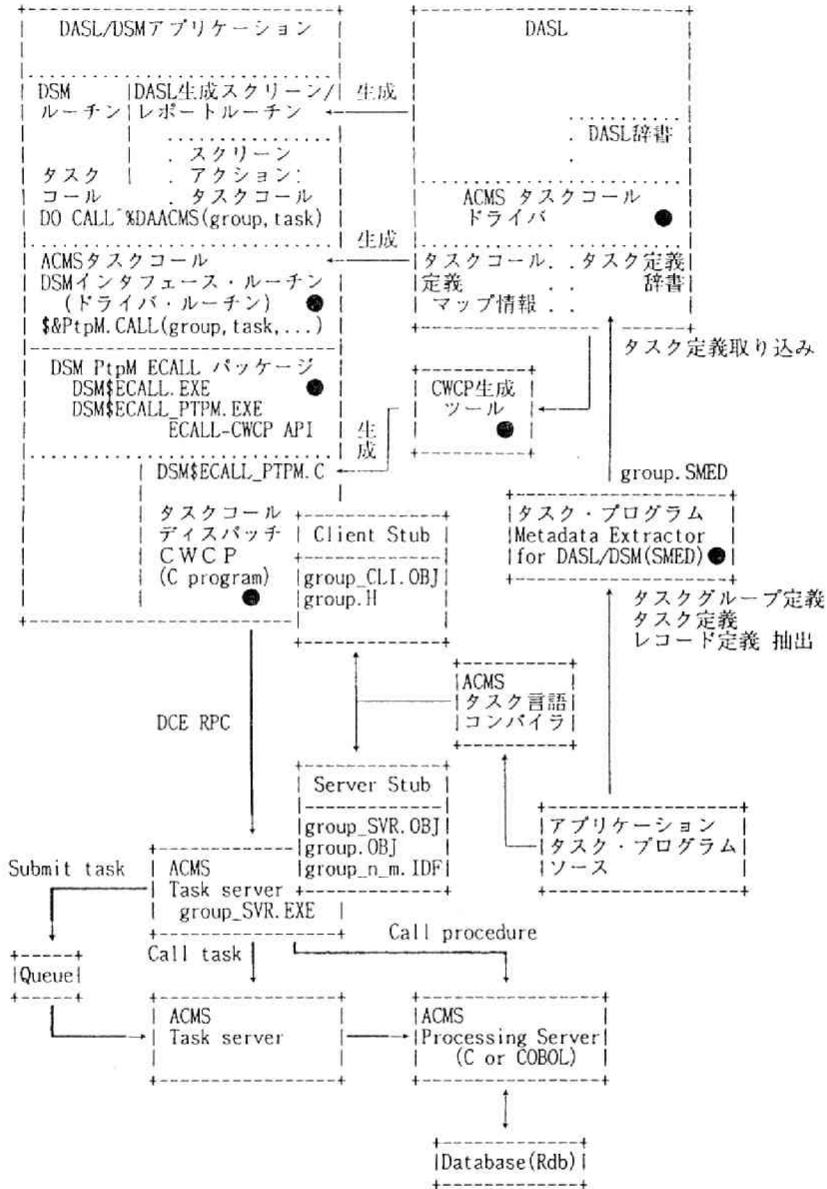
ンタフェース部プログラムの開発/保守性が向上します。

文末の図1に、DASL for ACMSの全体構造を示します。●が主要構成部です。

おわりに

MとM外の世界との統合上の試みとして、4GLアプローチによる事例を紹介しました。人間がある一定時間にタイプイン出来る量が限られている限り、プログラムの生産性を上げるには、少ないコマンドで多くのことが出来る必要があります。ひとつは、既にあるプログラムの再利用(サブルーチン)であり、ひとつは、上位抽象コマンドが多くのことを実行してくれる4GLアプローチです。この4GLアプローチの中で、M以外のシステム/ツールを統合利用する今回の試みは、Mの世界を広げるひとつの方向だと考えます。この種の試みはMの純血性を損なうかも知れません。しかし、なんでも自前でやるにはそのパワーが必要で、Mはそれができるか、それが賢明なのか問われています。Mは他との共存の中で、その特異性とテクノロジーを広め、深めて行くべきと考えます。そして、その時こそ、Mの真価が問われる時だと思えます。

図 1
DASL for ACMS の全体構造



最新 DSM の特徴と今後のプラン

Some features of the latest DSM and its future plan

佐藤比呂志

Hiroshi Sato

日本デジタルイクイップメント(株)
西日本第一統合システム部

Digital Equipment Corporation Japan
Enterprise Integration Center

〒530 大阪市中之島2丁目2番2号
ニチメンビル

TEL(06)-222-9211 FAX(06)-202-1419

要約

日本語 DSM for OpenVMS バージョン 6.2 が発表されました。バージョン 6.2 より Alpha AXP システム上でも DSM の稼動が可能となります。最新 DSM の特徴と今後のプラン、その中でも特に M WAPI(Window Application Interface) の実装について説明します。

We released Japanese DSM for OpenVMS Version 6.2. This version allows us to run DSM on the Alpha AXP system. This article describes some features of the new version, and also our plans on future DSM products especially M Window Application Programming Interface (MWAPI).

キーワード: DSM Alpha AXP MWAPI

最新 DSM の特徴

- 弊社の 64 ビット RISC ベース Alpha AXP システムで稼動する最初の DSM
- 1990 年度 TYPE A 言語拡張仕様の採用
- XMOTIF 外部呼び出し関数組み込み
- 新内部漢字コードの採用
- VAX DSM との相違点と VAX システムからの移行 Tool
- その他の機能強化点

今後の DSM プラン

DSM システムをサーバクライアントモデルに適合させるべく様々な製品が計画されています。

- SQLaccess for DSM
- M WAPI (M language Window Application Interface)
- 新しいプラットフォーム上での DSM
 - DSM for OSF/1 AXP
 - DSM for WNT
 - DSM for DOS
- 新しい DSM 用ネットワークプロトコル DDP+
- RPC (リモートプロシジャコール) 機構の組み込み

M WAPI の概要

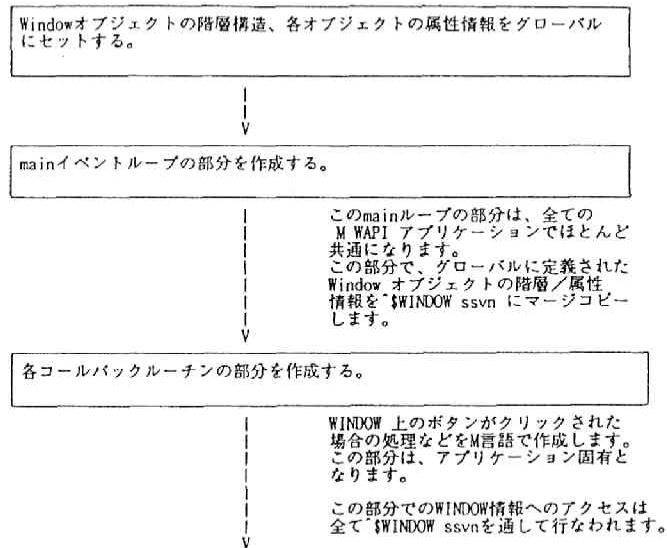
今後、Mでのウインドウプログラミングを可能とする M WAPI(Window Application Programming interface)について、DSM バージョン 6.2 での先行的実装例で説明します。

ssvn (System Structured Variable Names) の使用

Window オブジェクトの階層／属性等を表現する為に`^$WINDOW ssvn`を使用します。ssvn を利用することにより Window オブジェクトの複雑な階層構造／属性情報を柔軟に管理できます。

図 1

M WAPI でのプログラミング作業の流れ



この WAPI を利用して書かれたサンプルアプリケーション M 言語ソースプログラム、WINDOW 構造を定義するグローバルの一部を添付します。

main ルーチンの部分と一部のコールバックルーチンのリストです。

```

BURGER
BURGER ;PJ; Fast Food Application demo

GO ; main functin
D INIT
M ^$W("W1")=^BURGER("W1") ;load window info
D ^!ESTART ; enter event loop
K ^$W
Q

INIT ; initialize
K
K ^$W
Q

;***** Callback Subroutines *****

CHECK(REASON,AMOUNT) ; checkbox callback
S AMOUNT=$G(AMOUNT)
S GADGET=$P(REASON," ",1),REASON=$P(REASON," ",2)
W !,"You're in CHECK^BURGER; reason=",REASON,"", gadget=",GADGET,"", amount=",AMOUNT
I AMOUNT D TOTAL(AMOUNT)
Q

DOC(REASON) ; Document callback
S GADGET=$P(REASON," ",1),REASON=$P(REASON," ",2)
W !,"You're in DOC^BURGER; reason=",REASON,"", gadget=",GADGET
Q

LIST(REASON) ; list callback
S GADGET=$P(REASON," ",1),REASON=$P(REASON," ",2)
W !,"You're in LIST^BURGER; reason=",REASON,"", gadget=",GADGET
I REASON="SELECT" D
. ;non-standard, to be replaced per spec
. S ITEM=^!WEVENT($J,"CHOICE")
. S TEXT=^$W("W1","G","LIST","CHOICE",ITEM)
. S COST=$P(TEXT," ",!L(TEXT," "))
. I $D(^$W("W1","G","LIST","VALUE",ITEM)) S SIGN=1
. E S SIGN=-1
. ;end of non-standard code
. S TOTAL=$G(TOTAL)+(COST*SIGN)
. S ^$W("W1","G","TOTAL","VALUE")=$J(TOTAL,0,2)
. S ^$W("W1","G","S1","VALUE")=TOTAL*100
Q

LISTB(REASON) ; Listbutton callback
S GADGET=$P(REASON," ",1),REASON=$P(REASON," ",2)
W !,"You're in LISTB^BURGER; reason=",REASON,"", gadget=",GADGET
Q

LISTE(REASON) ; list entry box callback
S GADGET=$P(REASON," ",1),REASON=$P(REASON," ",2)
W !,"You're in LISTE^BURGER; reason=",REASON,"", gadget=",GADGET
Q

```

Window 構造を定義するグローバルの一部です。

```

^BURGER
^BURGER("W1") = Test Window Specification
^BURGER("W1", "EVENT", "CLICK") = WINDOW^BURGER("W1 CLICK")
^BURGER("W1", "EVENT", "FKEYDOWN") = WINDOW^BURGER("W1 FKEYDOWN")
^BURGER("W1", "EVENT", "FOCUS") = WINDOW^BURGER("W1 FOCUS")
^BURGER("W1", "EVENT", "PDOWN") = WINDOW^BURGER("W1 PDOWN")
^BURGER("W1", "EVENT", "PDRAG") = WINDOW^BURGER("W1 PDRAG")
^BURGER("W1", "EVENT", "PUP") = WINDOW^BURGER("W1 PUP")
^BURGER("W1", "EVENT", "UNFOCUS") = WINDOW^BURGER("W1 UNFOCUS")
^BURGER("W1", "G", "CHECK1", "ACTIVE") = 1
^BURGER("W1", "G", "CHECK1", "EVENT", "CHANGE") = CHECK^BURGER("CHECK1 CHANGE")
^BURGER("W1", "G", "CHECK1", "EVENT", "FOCUS") = CHECK^BURGER("CHECK1 FOCUS")
^BURGER("W1", "G", "CHECK1", "EVENT", "HELP") = CHECK^BURGER("CHECK1 HELP")
^BURGER("W1", "G", "CHECK1", "EVENT", "SELECT") = CHECK^BURGER("CHECK1 SELECT", .5)
^BURGER("W1", "G", "CHECK1", "EVENT", "UNFOCUS") = CHECK^BURGER("CHECK1 UNFOCUS")
^BURGER("W1", "G", "CHECK1", "POS") = 305,135
^BURGER("W1", "G", "CHECK1", "TFFACE") = M, FIXED
^BURGER("W1", "G", "CHECK1", "TFSTYLE") = BOLD
^BURGER("W1", "G", "CHECK1", "TITLE") = Fry .50
^BURGER("W1", "G", "CHECK1", "TYPE") = CHECK
^BURGER("W1", "G", "CHECK1", "VALUE") = 0
^BURGER("W1", "G", "CHECK1", "VISIBLE") = 1
^BURGER("W1", "G", "CHECK2", "EVENT", "FOCUS") = CHECK^BURGER("CHECK2 FOCUS")
^BURGER("W1", "G", "CHECK2", "EVENT", "HELP") = CHECK^BURGER("CHECK2 HELP")
^BURGER("W1", "G", "CHECK2", "EVENT", "SELECT") = CHECK^BURGER("CHECK2 SELECT", 1)
^BURGER("W1", "G", "CHECK2", "POS") = 305,170
^BURGER("W1", "G", "CHECK2", "TFFACE") = M, FIXED
^BURGER("W1", "G", "CHECK2", "TFSTYLE") = BOLD
^BURGER("W1", "G", "CHECK2", "TITLE") = Burger 1.00
^BURGER("W1", "G", "CHECK2", "TYPE") = CHECK
^BURGER("W1", "G", "CHECK3", "EVENT", "FOCUS") = CHECK^BURGER("CHECK3 FOCUS")
^BURGER("W1", "G", "CHECK3", "EVENT", "HELP") = CHECK^BURGER("CHECK3 HELP")
^BURGER("W1", "G", "CHECK3", "EVENT", "SELECT") = CHECK^BURGER("CHECK3 SELECT", .5)
^BURGER("W1", "G", "CHECK3", "POS") = 305,205
^BURGER("W1", "G", "CHECK3", "TFFACE") = M, FIXED
^BURGER("W1", "G", "CHECK3", "TFSTYLE") = BOLD
^BURGER("W1", "G", "CHECK3", "TITLE") = Hotdog .50
^BURGER("W1", "G", "CHECK3", "TYPE") = CHECK
^BURGER("W1", "G", "DOC1", "ACTIVE") = 1
^BURGER("W1", "G", "DOC1", "EVENT", "FKEYDOWN") = DOC^BURGER("DOC1 FKEYDOWN")
^BURGER("W1", "G", "DOC1", "EVENT", "FKEYUP") = DOC^BURGER("DOC1 FKEYUP")
^BURGER("W1", "G", "DOC1", "EVENT", "FOCUS") = DOC^BURGER("DOC1 FOCUS")
^BURGER("W1", "G", "DOC1", "EVENT", "HELP") = DOC^BURGER("DOC1 HELP")
^BURGER("W1", "G", "DOC1", "EVENT", "KEYDOWN") = DOC^BURGER("DOC1 KEYDOWN")
^BURGER("W1", "G", "DOC1", "EVENT", "KEYUP") = DOC^BURGER("DOC1 KEYUP")
^BURGER("W1", "G", "DOC1", "EVENT", "UNFOCUS") = DOC^BURGER("DOC1 UNFOCUS")
^BURGER("W1", "G", "DOC1", "FRAMED") = 0
^BURGER("W1", "G", "DOC1", "POS") = 300,290
^BURGER("W1", "G", "DOC1", "SIZE") = 14,3,CHAR
^BURGER("W1", "G", "DOC1", "TFFACE") = M, SERIF
^BURGER("W1", "G", "DOC1", "TFSTYLE") = ITALIC
^BURGER("W1", "G", "DOC1", "TITLE") = INSTRUCTIONS doc. title:
^BURGER("W1", "G", "DOC1", "TYPE") = DOCUMENT
^BURGER("W1", "G", "DOC1", "VISIBLE") = 1

```

逆\$ORDER関数の効用

— 本院の入院診療録管理システムへの応用を例に

馬場謙介*, 石名田洋一*, 鈴木孝始**

国立埼玉病院臨床研究部*, 同医事課**

本院の入院カルテの管理は、1989年からSP-MUMPS(住友電工)で実施してきた。当初は、逆\$ORDER関数がなかったので、逆\$ORDER関数に疑似的に等価なスモールコンセプトを作り対応していたが、逆\$ORDER関数が装備されたのを機会にシステムを書き換えたので、新旧を比較しながら逆\$ORDER関数の効用について報告する。

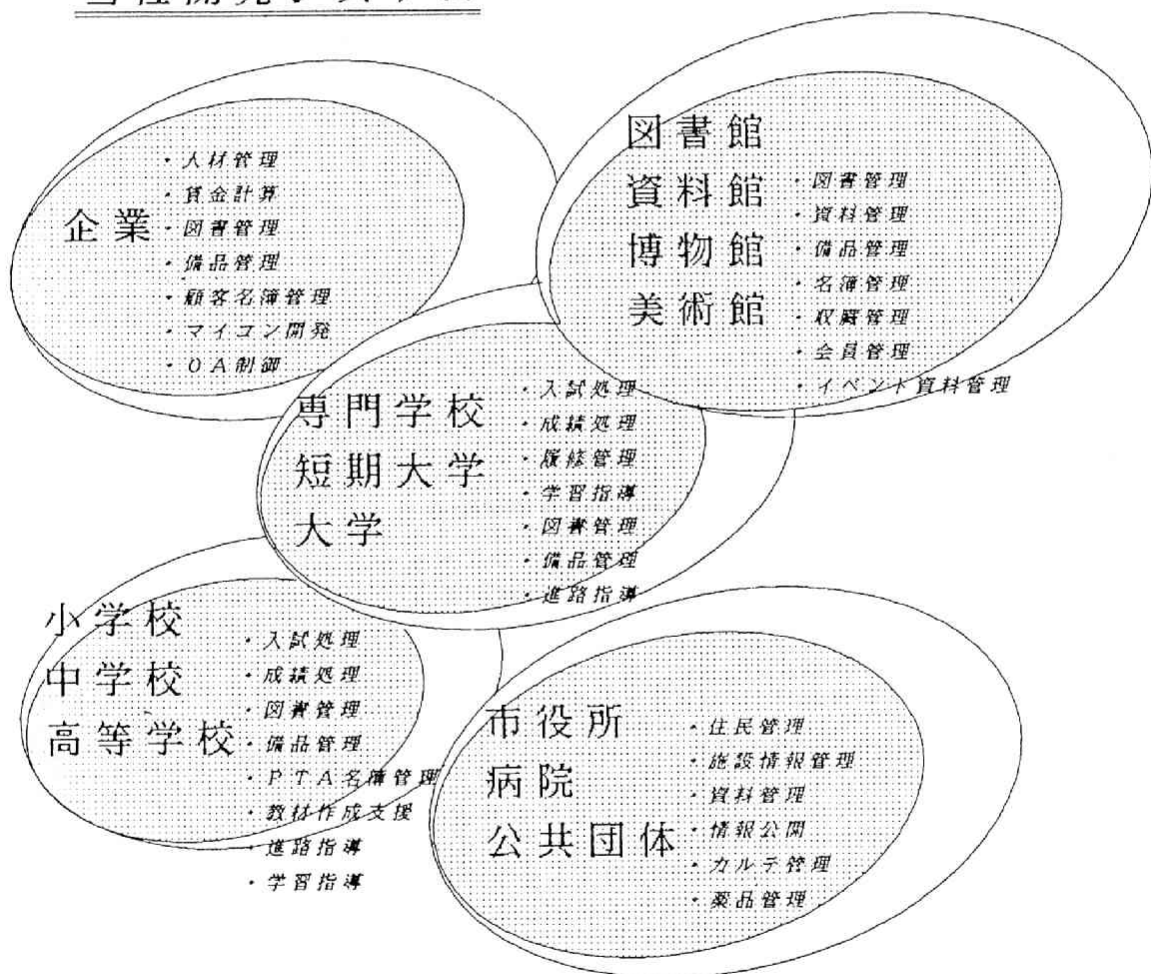
●教務事務作業を強力に支援する

総合成績管理システム

●新しい情報戦略のキーステーションを創造する!

図書・資料管理システム TNLIB Ver.2

当社開発システム



株式会社 高崎共同計算センター

事業本部 〒370 高崎市栄町4-11 第2原ビル 0273(27)2021(代) FAX.0273(27)3483
九段センター 〒101 千代田区飯田橋1-12-15 福岡第4ビル 03(3221)0901(代) FAX.03(3221)0903



病院経営の基盤安定には
最適なシステムが必須です。
だから、アクセル

アクセル効果①

病院全体の業務効率を高め、患者の待ち時間短縮を実現(サービス向上)

病院の各部門で発生した情報の加工・追加・変換・時間管理などを、利用する人のニーズに合わせて自動的に行ないます。これにより業務効率を高めるとともに、待ち時間の短縮をはじめとする患者サービスの向上を実現します。

アクセル効果②

患者情報、経営情報を一元管理(情報の有効利用)

各部門で入力されたすべての情報を蓄積、患者ごとに一元管理し、診断や経営戦略に役立つ各種情報に整理できます。その情報は部門ごとに設計された入力画面でマウスなどパソコン端末のマンマシン機能を使って容易に共同利用できます。

アクセル効果③

システム化の規模に合わせ段階的に導入(経済性)

システム化の規模や部門業務の増加に合わせ、段階的にアクセルパッケージ及び世界標準UNIXコンピュータなどのハードウェアから選択できる分散処理方式です。つまり「最初の一時期に必要な投資額の削減」が図られ、システムの効率的運用を実現します。

アクセル効果④

ドクターの診療判断や研究のサポート役として(機能の充実)

一元管理された患者情報から基本属性や診療歴の即時表示、検査結果変動分析などの診療支援システムのほか統計解析などの研究支援システムが充実しています。

アクセル効果⑤

ネットワーク型システム(発展性)

さまざまな検査機器やすでに導入済みのコンピュータと接続し、異機種間ネットワークを実現。さらに、病院全体のLANを構築し、統合情報ネットワークへと発展できます。

病院総合情報システム

病院経営を強力にサポートします。

ACCEL

ネットワーク型病院/診療所総合情報システム—アクセル

◆ 住友電気工業株式会社

情報通信システム事業部 MEシステム部システム営業課

◆ 住友電工システムズ株式会社

応用システム事業部

(お問合せは、上記両社とも)

東京都港区元赤坂1-3-12 107 ☎(03)3423-5680(代表)

大阪市西区土佐堀1-2-37 幸福ビル11F 〒550 ☎(06) 447-7151(代表)

名古屋市東区東桜1-1-6 住友商事ビル 〒461 ☎(052)963-2755(代表)

DTMってなあに？



パソコン用 M 言語総合開発システム

マルチユーザーマルチタスクの分散データベースシステム

驚異の実行速度、LAN 対応！



あなたに、何故 DTM をお勧めするのでしょうか？

- ◆ アプリケーションソフトウェア、ハードウェア資源を将来に渡って無駄にしません
- ◆ スクリーンエディター、動的ブレーク点、スタック表示、自動エラーロギングを含む総合開発環境です
- ◆ LAN の性能を引き出すサーバー・クライアント方式の分散処理システムです
- ◆ パソコンシステムなので、他の OS 環境よりもシステムの取り扱いか簡単です

総輸入元



D A T A T R E E
A Division of InterSystems Corporation

- ◆ パソコン用高性能 M 言語処理系
DataTree M 言語 (DTM-PC, DT-MAX)
- ◆ オプション
DT Windows (MS-Windows3.0以上に対応)
- ◆ 対応機種
DOS/V 対応パソコン
各社 AX 規格パソコン
NEC PC9801 シリーズ (一部制限あり)

日本ダイナシステム 株式会社

Japan DynaSystems Inc.

〒460 名古屋市中区新栄2-1-9 雲竜ビル東館504
Tel. 052-242-5441 Fax. 052-242-5984

NIFTYServe GGG03270 JDS



21世紀への計画はされていますか？
それなら、弊社が今提供できるものをご覧ください。

Alpha AXPシステム上で動作するDSM

弊社は業界をリードする新しい **Alpha AXPアーキテクチャ**

——明日のコンピュータパワーを取り込む最初の情報テクノロジー——
によってあなたを21世紀に導きます。

他に類を見ないパフォーマンス、価格性能比そしてアーキテクチャの持続性により、弊社の64ビット、RISC型のAlpha AXPアーキテクチャは、ソフトウェアの開発、実行環境の為に真に最初の普遍的なプラットフォームを提供します。

そしてM言語ユーザーを未来に導くため、21世紀最初のM言語の実装をご紹介します。

それが **DSM for OpenVMS** です。

DSM for OpenVMSは、従来DSMが、強力なコンピュータパワーを必要とする分散環境に対していつも提供していたのと同じ高可用性、高信頼性を提供します。

さらに、DSMソフトウェアは、**VAXシステム**と**Alpha AXPシステム**間で完全に**互換性**があります。つまりVAXシステム上で動作しているアプリケーションは何の修正もなくAlpha AXPシステム上で動作します。(日本語のごく一部に、互換性がない部分があります。) DSM for OpenVMSを使用して開発したアプリケーションは、開発コストを増やすことなく複数プラットフォームで動作させることができます。

いつAlpha AXPシステム用のDSM for OpenVMSが使用できるか？

それはいますぐにでも利用できます。21世紀はじめまで1秒たりとも待つ必要はありません。

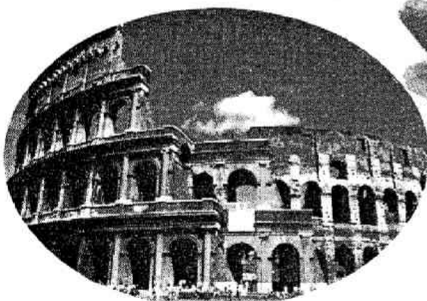
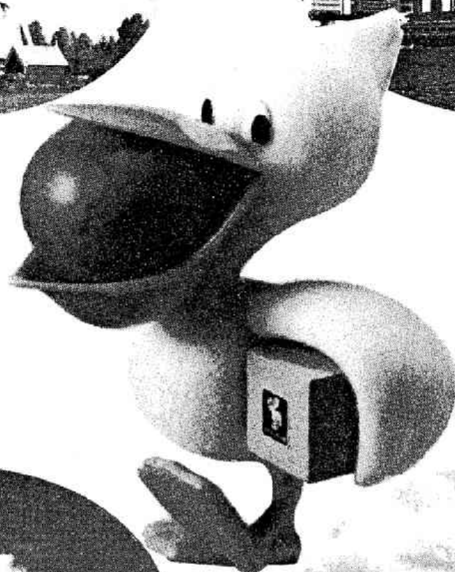
弊社の最新のM言語技術に関するご質問は、最寄りの日本DEC営業所、もしくは下記へご連絡ください。

東京：03-5992-7054

大阪：06-222-9211

digital

心で理が 日通海外引越 ハートライナー



大切なお荷物のエグゼクティブクラス



東京海外引越支店	(03)3572-4301	首都圏営業センター	(03)5400-2822
日本橋海外引越支店	(03)3272-5071	名古屋国際・海外引越課	(062)651-2621
新宿海外引越支店	(03)5379-1201/1202	大阪海外引越支店	(06)202-7671
五反田海外引越支店	(03)5496-9021/9121	神戸営業センター	(078)871-6561
横浜海外引越支店	(045)201-6975	京都営業センター	(075)352-2770

主な海外拠点

■ AMSTERDAM 020-6918819	■ DALIAN 0411-288888	■ JAKARTA 21-5706210	■ MUNCHEN 089-9050010	■ SEOUL 02-753-2691
■ ANTWERP 3-2339842	■ DALLAS 817-267-0914	■ KAOHSIUNG 07-333-5311	■ NEW YORK 718-932-4653	■ SHANGHAI 21-433-2398
■ ATLANTA 404-996-0702	■ DETROIT 313-941-5884	■ KUALA LUMPUR 3-3672194	■ PARIS 1-4894-5883	■ SINGAPORE 565-8177
■ AUCKLAND 9-256-0340	■ DUSSELDORF 0211-65570	■ LONDON 71-237-4881	■ PEKING 01-501-0530	■ STOCKHOLM 8-20-1027
■ BANGKOK 02-513-9449	■ FRANKFURT 06107-7740	■ LOS ANGELES 310-532-6300	■ PENANG 4-848822	■ SYDNEY 02-318-2422
■ BARCELONA 3-474-2828	■ GENEVA 022-788-0655	■ MADRID 1-674-0116	■ PHILADELPHIA 215-461-4300	■ TAIPEI 2-702-1161
■ BERLIN 030-3133006	■ GUANGZHOU 20-334-5239	■ MELBOURNE 03-330-3033	■ PORTLAND 503-281-6400	■ TORONTO 416-458-1008
■ BOSTON 617-569-7770	■ HAMBURG 040-73112-0	■ MEXICO CITY 5-211-5580	■ ROME 6-4880989	■ VANCOUVER 604-278-6084
■ BRUSSELS 2-751-7814	■ HONG KONG 408-1177	■ MIAMI 305-592-6109	■ SAN FRANCISCO 415-467-0100	■ WASHINGTON 301-725-0720
■ BUSAN 51-642-5136	■ HONOLULU 808-833-0202	■ MILANO 02-95343229	■ SAO PAULO 011-35-3011	■ WEIN 0222-71110-5411
■ CHICAGO 708-350-0202	■ HOUSTON 713-987-2300	■ MONTREAL 514-631-5436	■ SEATTLE 206-241-0885	■ ZURICH 01-836-9966

第20回日本エム・テクノロジー学会大会
予稿集
平成5年9月17日

編集 第20回日本エム・テクノロジー学会大会
大会長 山本 和子

発行 日本エム・テクノロジー学会 (MTA JAPAN)

日本MTA事務局
〒606-01
京都市左京区聖護院川原町54
京都大学医学部附属病院
医療情報部内